AS/400 Performance Capabilities Reference
Version 4 Release 3

Document Number ZC41-0607

September 1998

This document is intended for use by qualified performance related programmers or analysts from IBM, IBM Business Partners and IBM customers using AS/400e series. Information in this document may be readily shared with IBM AS/400 customers to understand the performance and tuning factors in OS/400 Version 4 Release 3. Requests for use of performance information by the technical trade press or consultants should be directed to Systems Performance Department V3T, IBM Rochester Lab, in Rochester, MN. USA.

Note!

Before using this information, be sure to read the general information under ″Special Notices″.

Eleveth Edition (September 1998)

This edition applies to Version 4, Release 3 of the AS/400 Operating System.

You can request a copy of this document by entering the following command on VM from MKTTOOLS. You may receive the LIST3820 format to print on a VM printer or a BOOK version. To request a copy of this for V4R3 (also applicable for RISC releases back to V3R6):

TOOLCAT MKTTOOLS GET AS4PPCPF PACKAGE

This Version 3 Release 3 Performance Capabilities Guide is available on the IBM AS/400 Internet site in the ″Reference Library″, at: http://www.as400.ibm.com. The document is also available on the IBM Intranet at: http://w3.rchland.ibm.com:80/cgi-bin/bookmgr/BOOKS/V4R3PERF/CCONTENTS

To request the CISC version (V3R2 and earlier), enter the following command on VM:

REQUEST V3R2 FROM FIELDSIT AT RCHVMW2 (your name

To request the IBM AS/400 Advanced 36 version, enter the following command on VM:

TOOLCAT MKTTOOLS GET AS4ADV36 PACKAGE

Contents

Special Notices

DISCLAIMER NOTICE

Performance data in this document was obtained in a controlled environment with specific performance benchmarks and tools. This information is presented along with general recommendations to assist the reader to have a better understanding of IBM(*) products. Results obtained in other environments may vary significantly and does not predict a specific customer′s environment.

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

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Commercial Relations, IBM Corporation, Purchase, NY 10577.

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

The following terms, which may or may not be denoted by an asterisk (*) in this publication, are trademarks of the IBM Corporation.

AS/400
C/400
OS/400
PS/2
OS/2
DB2
AFP
IBM
SQL/DS
400
CICS
S/370
RPG IV
AIX
IPDS
System/370
AS/400e
COBOL/400
RPG/400
CallPath
DRDA
SQL/400
ImagePlus
VTAM
APPN
SystemView
ValuePoint
DB2/400
ADSM/400
AnyNet/400
Operating System/400
AS/400e series
Application System/400
OfficeVision
Facsimile Support/400
Distributed Relational Database Architecture
Advanced Function Printing
Operational Assistant
Client Series
Workstation Remote IPL/400
Advanced Peer-to-Peer Networking
OfficeVision/400
AS/400 Advanced Application Architecture
ADSTAR Distributed Storage Manager/400
IBM Network Station

The following terms, which may or may not be denoted by a double asterisk (**) in this publication, are trademarks or registered trademarks of other companies as follows:

TPC Benchmark	Transaction Processing Performance Council
TPC-A, TPC-B	Transaction Processing Performance Council
TPC-C, TPC-D	Transaction Processing Performance Council
Lotus Notes, Lotus, Word Pro	Lotus Development Corporation
Notes, 123, CC Mail, Freelance	Lotus Development Corporation
Microsoft, Windows 95	Microsoft Corporation
Windows 95, Windows 95 Explorer	Microsoft Corporation
Microsoft Word, PowerPoint, Excel	Microsoft Corporation
ODBC, Windows NT Server, Access	Microsoft Corporation
Visual Basic, Visual C++	Microsoft Corporation
Adobe PageMaker	Adobe Systems Incorporated
Borland Paradox	Borland International Incorporated
CorelDRAW!	Corel Corporation
dBASEIII Plus	Borland International
Paradox	Borland International
WordPerfect	Satelite Software International
BEST/1	BGS Systems, Inc.
NetWare	Novell
Compaq	Compaq Computer Corporation
Proliant	Compaq Computer Corporation
BAPCo	Business Application Performance Corporation
Harvard Graphics	Software Publishing Corporation
HP-UX	Hewlett Packard Corporation
HP 9000	Hewlett Packard Corporation
INTERSOLV	Intersolve, Inc.
Q+E	Intersolve, Inc.
Netware	Novell, Inc.
Pentium	Intel Corporation
SPEC	Systems Performance Evaluation Cooperative
UNIX	UNIX Systems Laboratories
WordPerfect	WordPerfect Corporation
Powerbuilder	Powersoft Corporation
SQLWindows	Gupta Corporation
NetBench	Ziff-Davis Publishing Company
DEC Alpha	Digital Equipment Corporation
Java	Sun Microsystems, Inc.

Other terms that are used in this document may be trademarks of other companies.

Purpose of this Document

The intent of this document is to provide guidance in terms of AS/400 performance, capacity planning information, and tips to obtain best performance. This document is typically updated with each new release or more often if needed.

The V4R3 version of this document, dated September 1998, includes updates since the V4R2 version, dated February 1998.

For performance information on AS/400 Upgrade to RISC, refer to the document *AS/400 Upgrade Timing Information* that is available on MKTTOOLS. The package name is AS400TIM.

The wide variety of applications available makes it extremely difficult to describe a ″typical″ workload. The data in this document is the result of measuring or modeling certain application programs in very specific and unique configurations, and should not be used to predict specific performance for other applications. The performance of other applications can be predicted using a system sizing tool such as BEST/1(**) for OS/400 (refer to Appendix B, "AS/400 Sizing Tools" on page 313 for more details on BEST/1 support).

Related Publications / Documents

The following publications/documents are considered particularly suitable for additional information on AS/400 performance topics.

- *AS/400 Programming: Work Management Guide*, SC41-4306
- *AS/400 Programming: Performance Tools/400 Guide*, SC41-8084
- *AS/400 Performance Management V3R1*, GG24-3723-02
- The presentation *AS/400 Versus Microsoft′s SNA Server Gateway*

  This presentation and script discusses the advantages and disadvantages of the SNA Server approach. Included are the results of an IBM benchmark that compares the SNA Server to IBM direct attachment. To receive this package type the following on the PROFS command line:

  ```
  TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET AS4VSSNA PACKAGE
  ```

  If you are unable to access the information indicated above, please contact your IBM technical representative.

Chapter 1. Introduction

V4R3 continues to enhance the AS/400e series value proposition - the best melding of a superior operating system with 64-bit RISC hardware. The performance of V4R3 is greatly improved by both software enhancements and new hardware. AS/400e series continues to deliver customer usable performance in the multi-user, multi-applications environment by supporting interactive, client server, batch, groupware (Domino), Java, business intelligence, and web (e-business) serving.

High-end 8 and 12-way processors were added in V4R1, increasing performance of AS/400e series by 386% more than that available in V3R7. V4R3 supports new 8-way and 12-way systems, servers and custom servers offering up to 94% more performance than V4R2 models. In addition, 5 new performance features are available for the new AS/400e server 170. These new models all use the new IBM PowerPC A50 processor chip which is faster than A35. These new models offer better mixed mode performance and significant price/performance improvements. There is now over a 300-fold range of scalabilty in performance from the smallest e-series model to the largest 12-way.

The primary V4R3 performance items are:

- New models, both entry point and high-end growth
- Industry leading Lotus Domino mail performance and better price/performance
- Improved V4R3 CPW values, faster processors, more memory and disk
- New higher-performance and higher capacity I/O tape, disk and optical
- Enhanced server model performance characteristics and algorithm
- Reduced storage cost with possible performance improvements using first-to-market integrated hardware data compression (IHDC) for both disk and tape compression
- Parallel save/restore performance improvements with hierarchical storage management support based on user-defined policies
- Support for IBM Versatile Storage Server for sharing disks
- Reduced IPL times, both normal and abnormal IPLs
- Improve server-side Java performance
- Parallel data load and index maintenance for faster databases
- Improved Query performance on n-ways with encoded vector indexes
- Web server and secure web server performance improvements
- Reduced communications error recovery times

Customers who wish to remain with their existing hardware but want to move to the V4R3 operating system may also find improvements. In many cases, N-way systems that are limited due to high CPU utilization or high contention in V3R7 will see significant improvements with V4R3.

Version 4 Releases 1,2,and 3 of OS/400 continue to protect the customer′s investment while providing more function, growth, capacity, performance and better price/performance over previous versions. See the detailed sections of this document for further information.

Chapter 2. AS/400 System Capacities and CPW

This section contains information detailing the system capacities based on a workload called Commercial Processing Workload (CPW). For a detailed description, refer to Appendix A, "CPW Benchmark Description" on page 307.

CPW values are relative system performance metrics and reflect the relative system capacity for the CPW workload. CPW values can be used with caution in a capacity planning analysis (e.g., to scale CPU-constrained capacities, CPU time per transaction). However, these values may not properly reflect specific workloads other than CPW because of differing detailed characteristics (e.g., cache miss ratios, average cycles per instruction, software contention).

Additional factors that determine what performance is achievable are: type and number of disk devices, number of work station controllers, amount of memory, system model, and the application being run.

Testing in the Rochester laboratory has shown that for systems executing CPW there is no significant difference in performance between V4R2 and V4R3. It is possible that customers who have a different environment or a workload different from CPW may notice performance differences between the two releases. It is recommended that for those customers, other sections of the Performance Capabilities Reference Manual (or other sizing and positioning documents) be used to estimate the impact of upgrading to the new release.

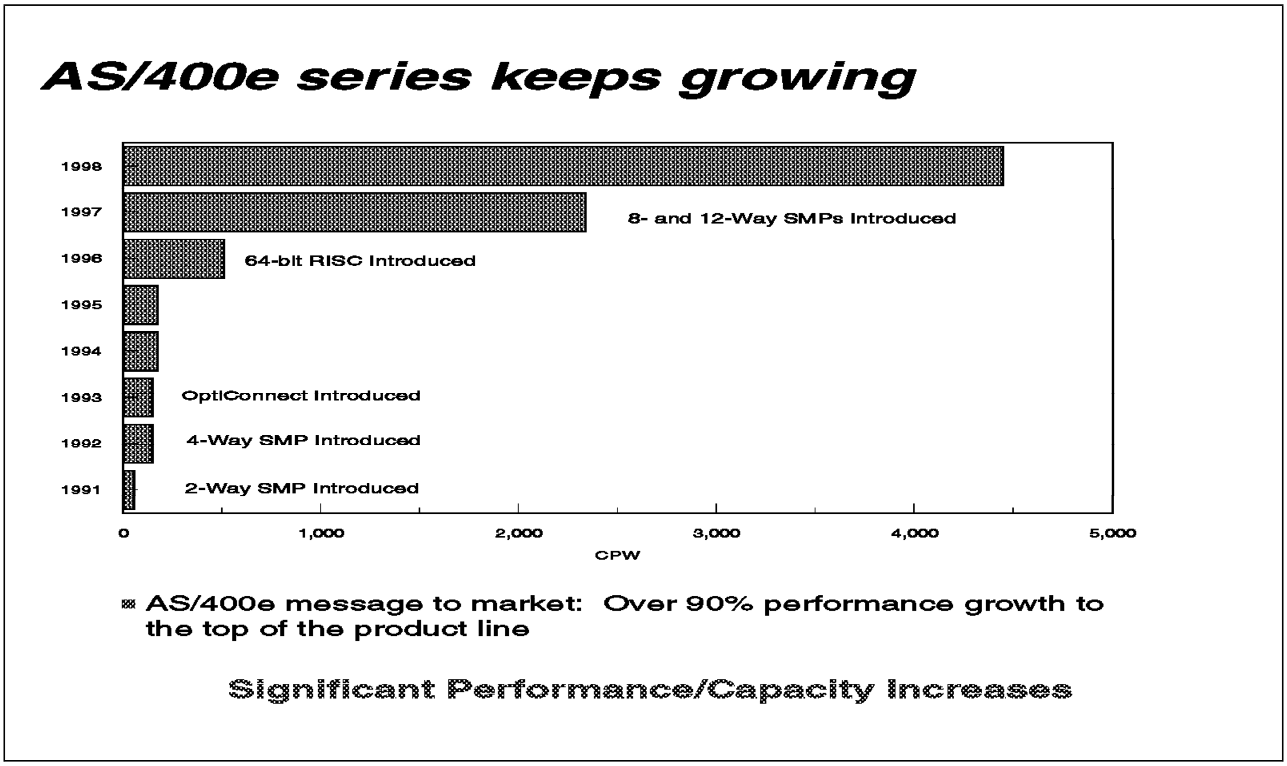
The tables that follow contain the CPW values for the AS/400e systems and servers. For additional CPW values, see the *IBM AS/400 Advanced 36 Performance Capabilities Reference*.

2.1 V4R3 additions

With the V4R3 additions of AS/400e System 650-2188 and 650-2189, AS/400e Server S40-2207 and S40-2208, and AS/400e Custom Server S40-2340 and S40-2341 some commercial environments could realize nearly double the maximum system capacity when compared to V4R2. This performance increase is the result of OS/400 improvements as well as higher processor, memory and DASD capacities. With the OS/400 improvements, customers with existing AS/400 systems may also see improved system capacities.

The environments where the most improvement has been observed are those with the following characteristics and using the following operating system functions.

- High CPU utilization
- High contention
- Commitment Control
- Large number of jobs
- Journaling

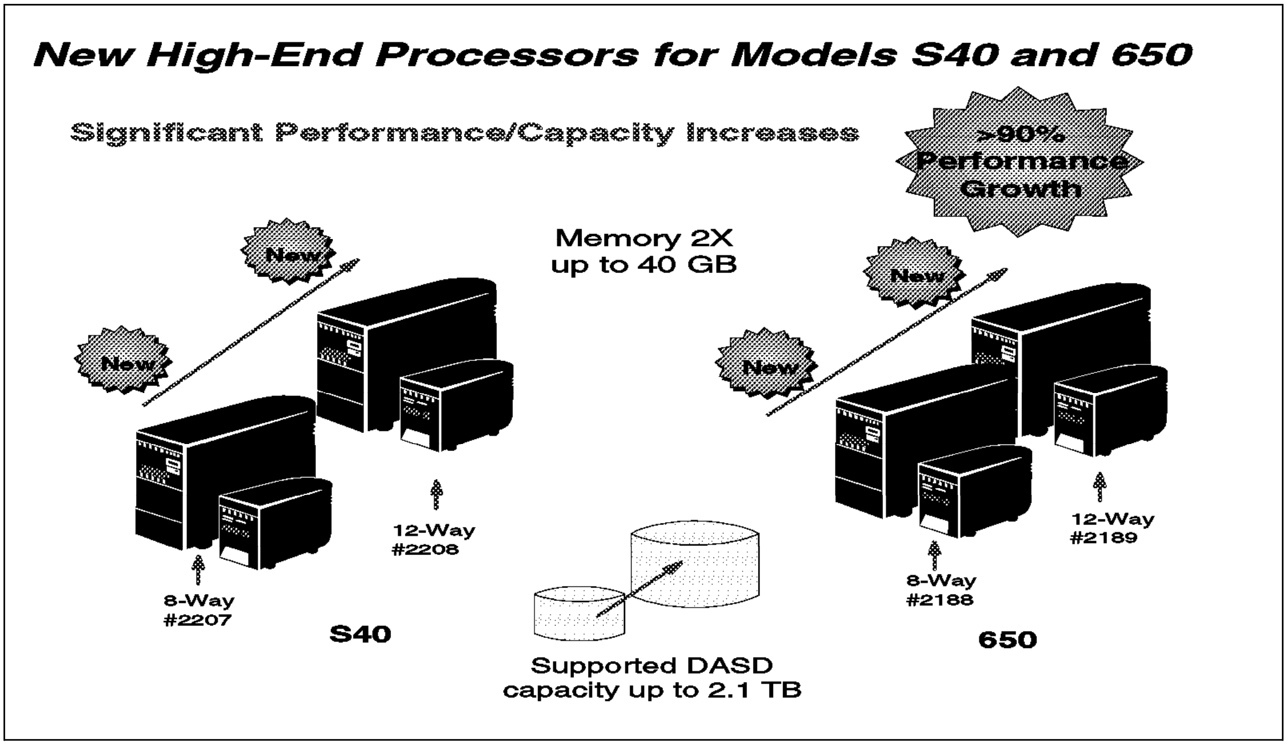


AS/400e series keeps growing

- AS/400e message to market: Over 90% performance growth to the top of the product line

Significant Performance/Capacity Increases

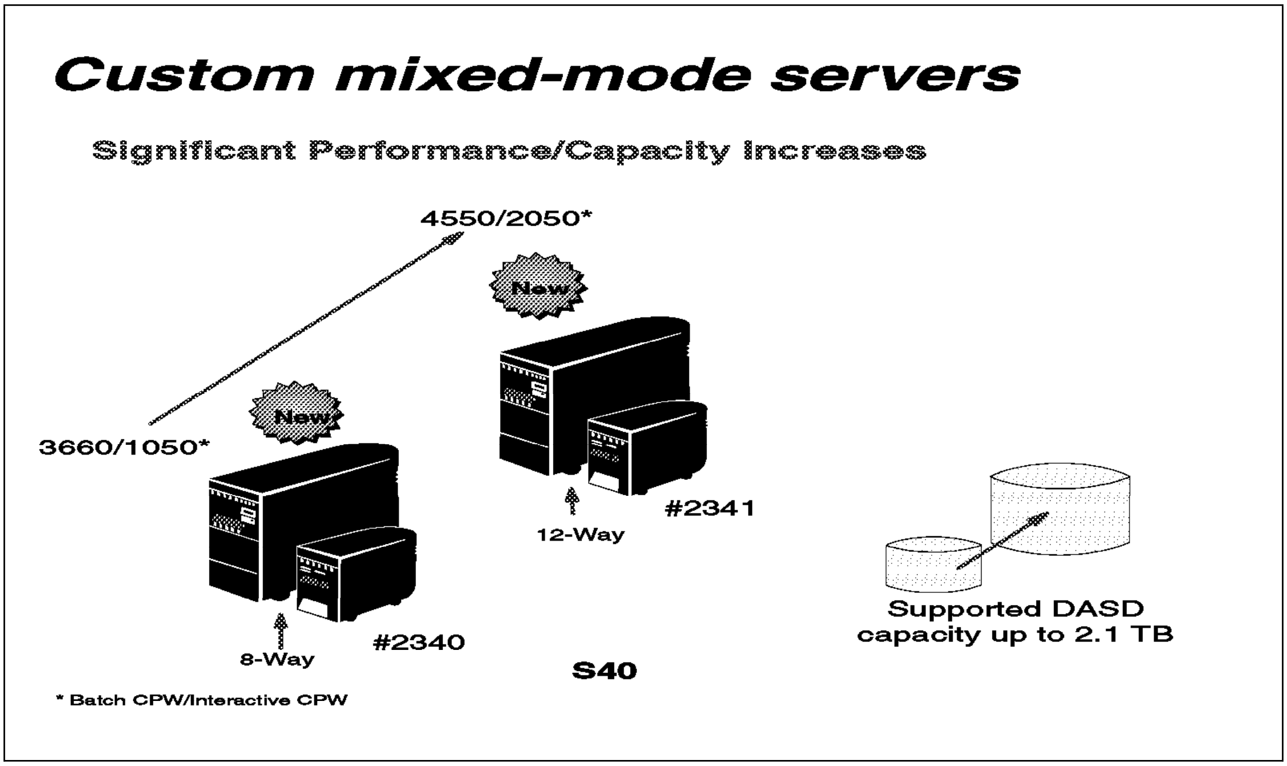
Figure 1.



New High-End Processors for Models S40 and 650

Significant Performance/Capacity Increases

Figure 2.

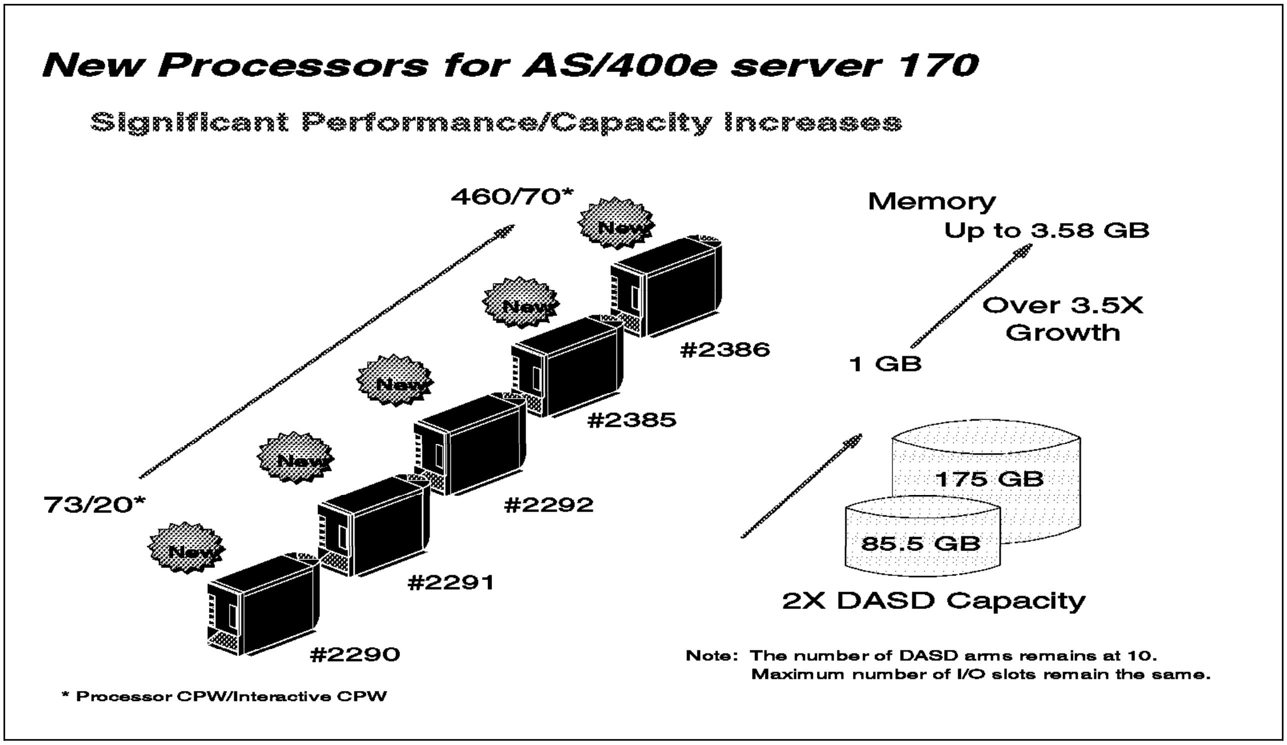


Custom mixed-mode servers

Significant Performance/Capacity Increases

* Batch CPW/Interactive CPW

Figure 3.



New Processors for AS/400e server 170

Significant Performance/Capacity Increases

Note: The number of DASD arms remains at 10. Maximum number of I/O slots remain the same.

* Processor CPW/Interactive CPW

Figure 4.

2.2 AS/400e systems

With the V4R3 additions of AS/400e System 650-2188 and 650-2189 some commercial environments could realize nearly double the maximum system capacity when compared to V4R2.

Table 1. Relative Performance of AS/400e Systems

Model	Feature Code	Memory Capacity (MB) Maximum	Disk Capacity (GB) Maximum	CPUs	V4R3 CPW
600	2129	384	175.4	1	22.7
	2134	384	175.4	1	32.5
	2135	384	175.4	1	45.4
	2136	512	175.4	1	73.1
620	2175	1856	944.8	1	50.0
	2179	2048	944.8	1	85.6
	2180	2048	944.8	1	113.8
	2181	2048	944.8	1	210.0
	2182	4096	944.8	2	464.3
640	2237	16384	1340	1	319.0
	2238	8704	1340	2	583.3
	2239	16384	1340	4	998.6
650	**2188**	40960	2095.9	8	3660
	2189	40960	2095.9	12	4550
	2240	32768	2095.9	8	1794
	2243	32768	2095.9	12	2340

2.3 AS/400e Servers

The new V4R3 additions are S40-2207 and S40-2208.

Table 2. Relative Performance of AS/400e Servers

Model	Feature Code	Memory Capacity (MB) Maximum	Disk Capacity (GB) Maximum	CPUs	V4R3 CPW
S10	2118	384	175.4	1	45.4/16.2
	2119	512	175.4	1	73.1/24.4
S20	2161	2048	944.8	1	113.8/31.0
	2163	2048	944.8	1	210.0/35.8
	2165	4096	944.8	2	464.3/49.7
	2166	4096	944.8	4	759.0/56.9
S30	2257	16384	1340	1	319.0/51.5
	2258	16384	1340	2	583.3/64.0
	2259	16384	1340	4	998.6/64.0
	2260	16384	1340	8	1794/64.0
S40	**2207**	40960	2095.9	8	3660/120
	2208	40960	2095.9	12	4550/120
	2256	32768	2095.9	8	1794/64.0
	2261	32768	2095.9	12	2340/64.0

Note: The first number in the CPW column represents the Client/Server or batch performance rating, while the second number represents the interactive performance rating.

2.4 AS/400e Custom Servers

The new V4R3 addition is the S40 model.

Table 3. Relative Performance of AS/400e Custom Servers

Model	Feature Code	Memory Capacity (MB) Maximum	Disk Capacity (GB) Maximum	CPUs	V4R3 CPW
S20 Custom	2177	4096	704	4	759.0/110.7
	2178	4096	704	4	759.0/221.4
S30 Custom	2320	8704	927.7	4	998.6/215.1
	2321	12288	927.7	8	1794/386.4
	2322	12288	927.7	8	1794/579.6
S40 Custom	**2340**	40960	2095.9	8	3660/1050
	2341	40960	2095.9	12	4550/2050

Note: The first number in the CPW column represents the Client/Server or batch performance rating, while the second number represents the interactive performance rating.

Consult configuration rules for special requirements on Custom models.

2.5 AS/400e Custom Application Server Model SB1

AS/400e application servers are particularly suited for environments with minimal database needs, minimal disk storage needs, lots of low-cost memory, high-speed connectivity to a database server, and minimal upgrade importance.

The SB1 throughput rates in terms of Financial (FI) dialogsteps per hour with measurements at 65% and 95% utilization points may be used by marketing to size systems marketing to size systems for customer orders.
Note: 1 SD ds == 2.5 FI ds. (SD = Sales & Distribution).

Table 4. AS/400e Custom Application Server Model SB1

Model	CPUs	SAP Release	SD ds/hr @ 65% CPU utilization	FI ds/hr @ 65% CPU utilization
2312	8	3.1H	109,770.49	274,426.23
		4.0B	65,862.29	164,655.74
2313	12	3.1H	158,715.76	396,789.40
		4.0B	95,229.46	238,073.64

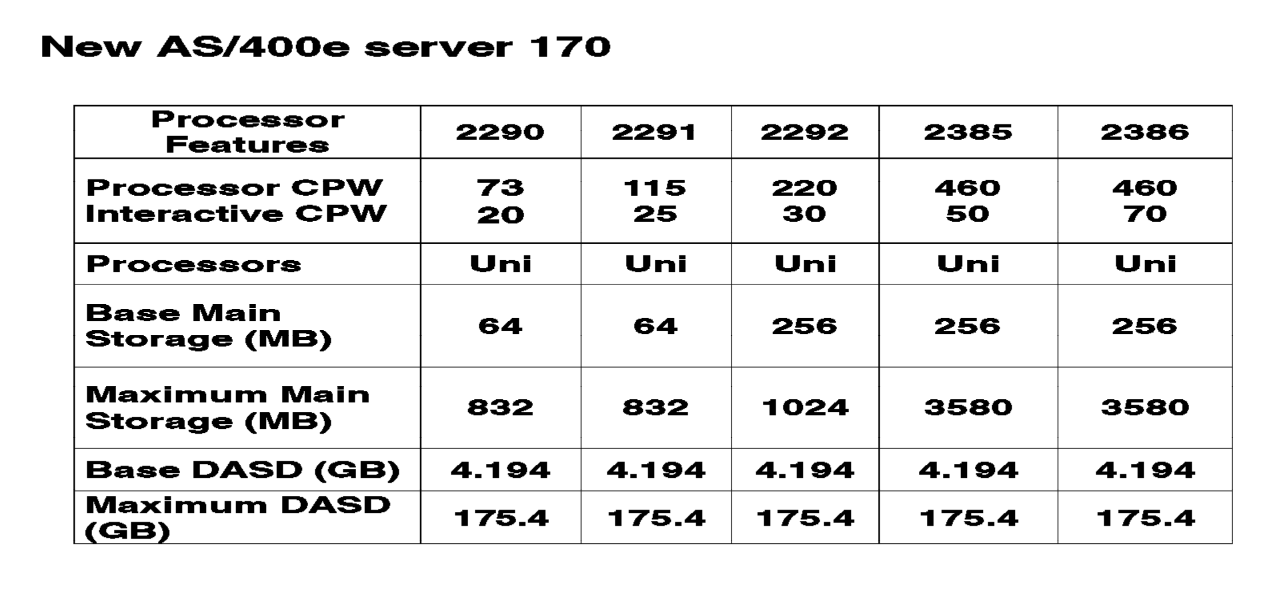
2.6 AS/400 e-server 170

Highlights of new model 170s when compared to previous models:

Improved server performance (up to 44% more processing capability) due to improved processor technology and the availability of additional memory. The most improvement will be seen in CPU intensive workloads. Workloads that require many disk drives and/or memory may not see as much gain. The server CPW will reflect processor capability. In the past, the server CPW reflected a system capability to exercise the CPW workload. Certain constraints such as DASD and memory may cause the processor capability to not be reached. Customer results will vary depending on workload characteristic.

The new 170 models provide improved interactive performance. With previous models, system overhead began to affect the overall capacity of the system when interactive requirements exceeded 1/3 of the interactive capacity. With the new models, this overhead does not occur until the announced interactive capacity is reached. This allows the new models to operate more smoothly in interactive and mixed environments. The absolute limit for interactive capacity on the new models can be computed by multiplying the interactive CPW rating by a factor of 7/6.

As with previous mixed-mode and server models, a system value (QDYNPTYADJ) is available to determine how the system will react to work requests when interactive exceeds the ″knee″ (point where overhead is introduced). If the system value is turned on, non-interactive work is favored over additional interactive work. If it is turned off, additional interactive work is allowed at the expense of low-priority non-interactive work. With previous model 170 systems, this value affected the system when interactive requirements exceeded 1/3 of the interactive capacity of the system. With the new systems, this value only affects the system when interactive requirements exceed the published performance ratings.



New AS/400e server 170

Processor Features	2290	2291	2292	2385	2386
Processor CPW Interactive CPW	73 20	115 25	220 30	460 50	460 70
Processors	Uni	Uni	Uni	Uni	Uni
Base Main Storage (MB)	64	64	256	256	256
Maximum Main Storage (MB)	832	832	1024	3580	3580
Base DASD (GB)	4.194	4.194	4.194	4.194	4.194
Maximum DASD (GB)	175.4	175.4	175.4	175.4	175.4

Figure 5.

Table 5. V4R2 AS/400 e-server 170 values

V4R2 Model 170	Feature Code	Memory Capacity (MB) Maximum	Disk Capacity (GB) Maximum	CPUs	V4R2 CPW
1/3 algorithm (Constrained)	2159	1024	85.8	1	73/16
	2160	1024	85.8	1	114/23
	2164	1024	85.8	1	125/29
	2176	1024	85.8	1	125/40
	2183	1024	85.8	1	125/67
1/3 algorithm (Unconstrained)	2159	1024	85.8	1	73/16
	2160	1024	85.8	1	114/23
	2164	1024	85.8	1	210/29
	2176	1024	85.8	1	319/40
	2183	1024	85.8	1	319/67

Note: The first number in the CPW column represents the Client/Server or batch performance rating, while the second number represents the interactive performance rating.

2.7 Pre-V4R3 AS/400 models - CPW ratings

The following charts are included for historical reasons.

AS/400 e-server model 150

The model 150 was last changed in V4R1 to allow additional memory to be configured. Feature code 2270 also received a modest performance boost (approximately 5%) in V4R1 with software improvements.

In the table below are entries for both constrained and unconstrained CPW values, in releases prior to V4R1 the system was constrained for a commercial environment due to the 92 MB memory limit. With the larger capacity for V4R1, memory is no longer the limiting resource. For V4R1, the limit of 4 DASD devices is the constraining resource. For workloads that do not perform as many disk operations or don′t require as much memory, the non-constrained CPW value may be more representative of the performance capabilities.

Table 6. Relative Performance of the AS/400 e-server 150

Model 150	**Feat. Code**	**Memory Capacity (MB) Maximum**		**Disk Capacity (GB) Maximum**	**V3R7 CPW**	**V4R1 CPW**
		V3R7	**V4R1**			
Constrained	2269	96	192	16.7	10.9/10.9	20.2/13.8
	2270	96	192	16.7	10.9/10.9	20.2/20.2
Unconstrained	2269	96	192	16.7	27.0/13.8	27.0/13.8
	2270	96	192	16.7	33.3/20.6	35.0/20.6

Note: The first number in the CPW column represents the Client/Server or batch performance rating, while the second number represents the interactive performance rating.

Previous AS/400 RISC Systems Capacities

Table 7. Relative Performance of previous AS/400 RISC Systems

Model	Feature Code	Memory (MB) Maximum	Disk (GB) Maximum	V3R6 CPW	V3R7 CPW	V4R1 CPW
400	2130	160	50	12.3	13.8	13.8
	2131	224	50	18.3	20.6	20.6
	2132	224	50	24.5	27.0	27.0
	2133	224	50	30.6	33.3	35.0
500	2140	768	652	18.7	21.4	21.4
	2141	768	652	26.9	30.7	30.7
	2142	1024	652	38.3	43.9	43.9
510	2143	1024	652	66.7	77.7	81.6
	2144	1024	652	85.0	104.2	111.5
530	2150	4096	996	107.1	131.1	148.0
	2151	4096	996	132.5	162.7	188.2
	2152	4096	996	198.7 2	278.8 2	319.0 2
	2153	4096	996	299.0 4	459.3 4	598.0 4
	2162	4096	996	349.8 4	509.9 4	650.0 4

Note: The n symbol indicates the number of CPU′s when greater than 1.

Previous AS/400e Server Model Capacities

Table 8. Relative Performance of AS/400 Servers

Model	Feature Code	Memory (MB) Maximum	Disk (GB) Maximum	V3R6 CPW	V3R7 CPW	V4R1 CPW
40S	2109	224	50	24.5/8.4	27.0/9.4	27.0/9.4
	2110	224	50	30.6/12.3	33.3/13.8	35.0/14.5
	2111	512	50	52.9/18.3	59.8/20.6	63.0/21.6
	2112	512	50	77.3/26.9	87.3/30.7	91.0/32.2
50S	2120	1024	652	66.7/18.7	77.7/21.4	81.6/22.5
	2121	1024	652	85.0/26.9	104.2/30.7	111.5/32.2
	2122	1024	652	106.8/26.9	130.7/30.7	138.0/32.2
53S	2154	4096	996	132.5/26.9	162.7/30.7	188.2/32.2
	2155	4096	996	198.7/26.9 2	278.8/30.7 2	319.0/32.2 2
	2156	4096	996	299.0/26.9 4	459.3/30.7 4	598.0/32.2 4
	2157	4096	996	349.8/26.9 4	509.9/30.7 4	650.0/32.2 4

Note: The n symbol indicates the number of CPU′s when greater than 1.
The first number in the CPW column represents the Client/Server or batch performance rating, while the second number represents the interactive performance rating.

AS/400 CISC Model Capacities

Table 9. Relative Performance of AS/400 Systems (9401)

9401 Model	Memory Capacity (MB) Maximum	Disk Capacity (GB) Maximum	CPW
P02	16	2.1	7.3
P03 2114	24	2.99	7.3
P03 2115	40	3.93	9.6
P03 2117	56	3.93	16.8

Table 10. Relative Performance of AS/400 Systems (9402)

9402 Model	Memory Capacity (MB) Maximum	Disk Capacity (GB) Maximum	CPW
C04	12	1.3	3.1
C06	16	1.3	3.6
D02	16	1.2	3.8
D04	16	1.6	4.4
E02	24	2.0	4.5
D06	20	1.6	5.5
E04	24	4.0	5.5
F02	24	2.1	5.5
F04	24	4.1	7.3
E06	40	7.9	7.3
F06	40	8.2	9.6

Table 11. Relative Performance of AS/400 Systems (9404)

9404 Model	Memory Capacity (MB) Maximum	Disk Capacity (GB) Maximum	CPW
B10	16	1.9	2.9
C10	20	1.9	3.9
B20	28	3.8	5.1
C20	32	3.8	5.3
D10	32	4.8	5.3
C25	40	3.8	6.1
D20	40	4.8	6.8
E10	40	19.7	7.6
D25	64	6.4	9.7
F10	72	20.6	9.6
E20	72	19.7	9.7
F20	80	20.6	11.6
E25	80	19.7	11.8
F25	80	20.6	13.7

Table 12. Relative Performance of AS/400 Systems (9406)

9406 Model	Memory Capacity (MB) Maximum	Disk Capacity (GB) Maximum	CPW
B30	36	13.7	3.8
B35	40	13.7	4.6
B40	40	13.7	5.2
B45	40	13.7	6.5
D35	72	67.0	7.4
B50	48	27.4	9.3
E35	72	67.0	9.7
D45	80	67.0	10.8
D50	128	98.0	13.3
E45	80	67.0	13.8
F35	80	67.0	13.7
B60	96	54.8	15.1
F45	80	67.0	17.1
E50	128	98.0	18.1
B70	192	54.8	20.0
D60	192	146	23.9
F50	192	114	27.8
E60	192	146	28.1
D70	256	146	32.3
E70	256	146	39.2
F60	384	146	40.0
D80	384	256	56.6 2
F70	512	256	57.0
E80	512	256	69.4 2
E90	1024	256	96.7 3
F80	768	256	97.1 2
E95	1152	256	116.6 4
F90	1024	256	127.7 3
F95	1280	256	148.8 4
F97	1536	256	177.4 4

Note: The n symbol indicates the number of CPU's when greater than 1.

Table 13. Relative Performance of AS/400 Advanced Systems (CISC)

Model	Feature Code	Memory Capacity (MB) Maximum	Disk Capacity (GB) Maximum	CPUs	CPW
200	2030	24	23.6	1	7.3
	2031	56	23.6	1	11.6
	2032	128	23.6	1	16.8
300	2040	72	117.4	1	11.6
	2041	80	117.4	1	16.8
	2042	160	117.4	1	21.1
310	2043	832	159.3	1	33.8
	2044	832	159.3	2	56.5
320	2050	1536	259.6	1	67.5
	2051	1536	259.6	2	120.3
	2052	1536	259.6	4	177.4

Table 14. Relative Performance of AS/400 Server Models (9402,9404)

Model	Feature Code	Memory Capacity (MB) Maximum	Disk Capacity (GB) Maximum	CPUs	CPW
9402	S01	56	3.9	1	17.1/5.5
	100	56	7.9	1	17.1/5.5
9404	135	384	27.5	1	32.3/9.6
	140	512	47.2	2	65.6/11.6

Note: The first number in the CPW column represents the Client/Server or batch performance rating, while the second number represents the interactive performance rating.

Table 15. Relative Performance of AS/400 Advanced Servers (CISC models)

Model	Feature Code	Memory Capacity (MB) Maximum	Disk Capacity (GB) Maximum	CPUs	CPW
20S	2010	128	23.6	1	17.1/5.5
2FS	2010	128	7.8	1	17.1/5.5
2SS	2010	128	7.8	1	17.1/5.5
2SG	2010	128	7.8	1	17.1/5.5
30S	2411	384	86.5	1	9.6/32.3
	2412	832	86.5	2	68.5/11.6

Note: The first number in the CPW column represents the Client/Server or batch performance rating, while the second number represents the interactive performance rating.

2.8 OS/400 RISC Server Model Behavior

Overview of OS/400 RISC Server Model Behavior

AS/400 Advanced Servers and AS/400e servers are intended for use primarily in client/server or other non-interactive work environments. 5250-based interactive work can be run on these servers, but with limitations.

The underlying performance structure of the AS/400e custom servers is the same as that of the AS/400 Advanced Servers and the AS/400e servers. AS/400e custom servers are designed to provide optimally balanced performance in non-interactive and interactive environments as required by specific ISV software. The primary environment addressed is one which runs significant interactive 5250 applications and is migrating toward a client/server ″graphical″ version of the 5250 applications as well as new client/server applications.

For the rest of this chapter, server is generically used to represent AS/400 Advanced Server, AS/400e server, custom server, and 170 server.

RISC server models have been available since V3R6. There is a basic server model behavior that applies to all server models which is described in the following section. There are also following sections that discuss server performance behaviors for the AS/400e custom server and AS/400e server 170.

5250-based interactive work (hereafter called interactive) is defined as any job doing 5250 display device I/O. This includes:

- All 5250 sessions
- Any green screen interface
- Telnet or 5250 DSPT workstations
- 5250/HTML workstation gateway
- PC′s using 5250 emulation
- Interactive program debugging
- PC Support/400 work station function
- RUMBA/400
- Screen scrapers
- Interactive subsystems
- Twinax printer jobs
- BSC 3270 emulation
- 5250 emulation

In general, the Type column of the Work with Active Jobs (WRKACTJOB) command will tell you which jobs are considered interactive (type = INT). These are all jobs that were initiated by signing on at a 5250 display device.

Another way to determine interactive work is to run the Performance Monitor (STRPFRMON, ENDPFRMON) during selected time periods and review the first page of the Performance Tools/400 licensed program Component Report.

Non-interactive work (hereafter called client/server) is defined to be any application that is not interactive such as batch, client/server, database, etc.

The RISC server models have been available since V3R6. There is a basic server model behavior that applies to all server models. Enhancements have been made that are only available in a specific version/release, as described below.

The performance information and equations mentioned here represent ideal environments. This information is presented along with general recommendations to assist the reader to have a better understanding of the AS/400e server models. Actual results may vary significantly.

Server Model Behavior

Traditional (non-server) AS/400 models have a single CPW value which represents the maximum workload that can be applied to that model. This CPW value is applicable to either an interactive workload, a client/server workload, or a combination of the two.

On a server or custom server model, there are two CPW values. The larger value represents the maximum workload the model could support if the workload were entirely client/server (i.e. no interactive components). The smaller CPW value represents the maximum workload the model could support if the workload were entirely interactive. **These values are NOT additive - interactive processing will reduce the system′s client/server processing capability.**

Custom server models interact to batch and interactive workloads similar to the server models but the degree of interaction and priority of workloads follows a different algorithm and hence the knee of the curve for workload interaction is at a different point (see Custom Server Behavior section for details).

Server models were designed for a client/server workload and to accommodate a small interactive workload. When the interactive workload exceeds approximately 1/3 (6/7 for AS/400e custom servers) of the maximum interactive CPW, the client/server processing performance of the system becomes increasingly impacted. Once the interactive workload reaches the maximum interactive CPW value, all the CPU cycles are being used and there is no capacity available for handling client/server tasks.

Microcode task CFINTnn, for all AS/400 models, is an indication of the time to handle interrupts and task switching. For the server models, it is also an indication of how much the system performance has been impacted by the interactive processing. There is one CFINTnn task for each processor. For example, on a single processor system only CFINT01 will appear. On an 8-way processor, system tasks CFINT01 through CFINT08 will appear.

The effective interactive utilization for a server or custom server system can be defined as the useable interactive utilization plus the total of CFINT utilization.

The following figure graphically represents the above for a model S20-2161. For this model, note that below about 9% (the ″knee of the curve″), the effective interactive utilization is roughly equal to the useable interactive utilization. Also note the substantially increased rate of total CPU utilization (due to interactive workload) once the useable interactive workload exceeds 9%. For example, with a useable interactive utilization of 20%, CFINT will indicate that an additional 43% (approximate) of the total CPU will be used, yielding an effective interactive CPU utilization of approximately 63%. This leaves approximately 37% of the total CPU available for client/server work. Finally, note that the CPU is completely utilized once the useable interactive workload reaches about 27% (CFINT would indicate approximately 73% CPU). At this saturation point, there is no CPU available for client/server.

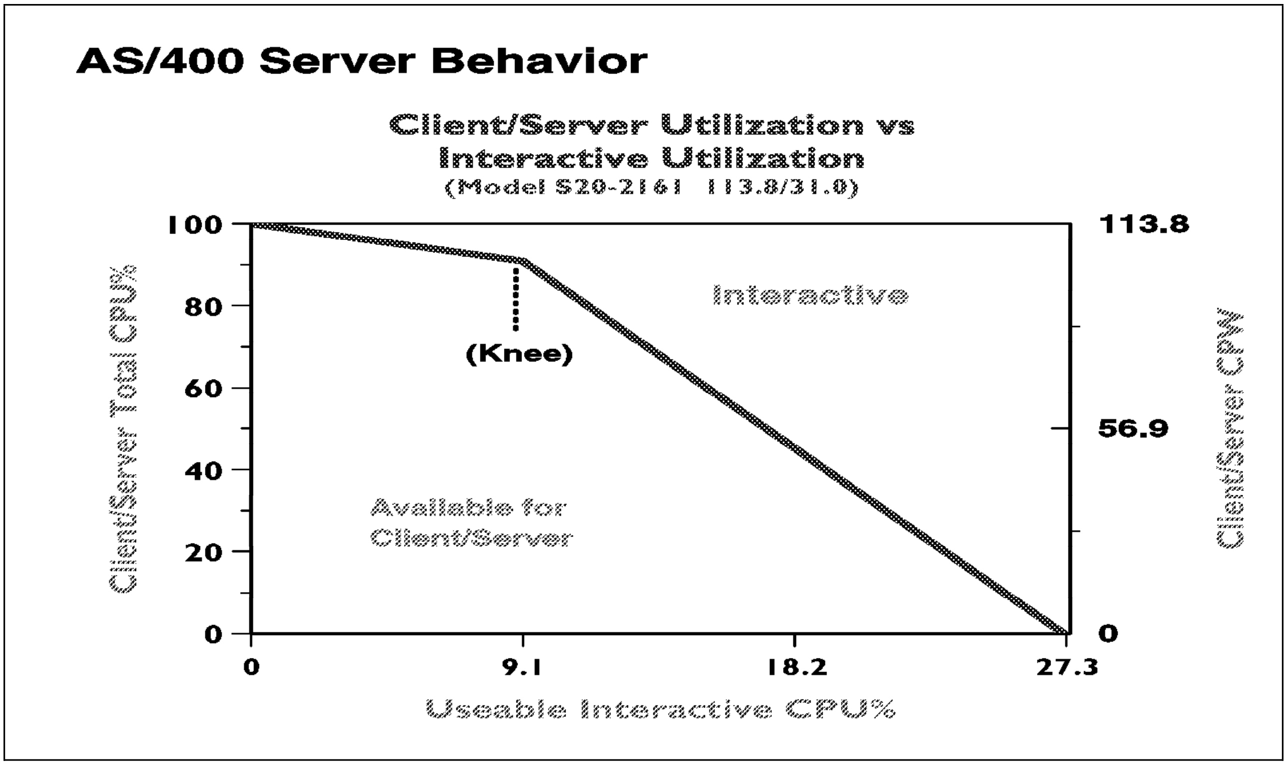


Figure 6.

Similar figures may be constructed for other server models by simply using the applicable values from the tables below for the knee of the curve and for the maximum interactive CPU.

Customers should plan for their interactive workloads to execute below (less than) the knee of the curve.

In general, executing a single interactive job will not cause a significant impact to client/server performance.

The figure above shows a straight line for the effective interactive utilization, which is the intent of the server behavior. Real/customer environments will produce a curved line since most environments will be dynamic, due to job initiation, interrupts, etc, etc.

The following tables show, for each server model, the maximum interactive CPW and its corresponding CPU % and the point (about 1/3 maximum interactive - the knee of the curve) where the interactive utilization begins to increasingly impact client/server performance. The knee of the curve is given in CPW values as well as CPU %. For those models that are n-ways (multiple processors), the percent value is the percent of all the processors (not of a single one).

AS/400e servers

Model	Feature	Maximum Client/Serv CPW	Maximum Interactive CPW	CPU % at Maximum Interactive	1/3 Max Interactive CPW (Knee)	CPU % at the Knee
S10	2118	45.4	16.2	35.7	5.4	11.9
	2119	73.1	24.4	33.4	8.1	11.1
S20	2161	113.8	31.0	27.2	10.3	9.1
	2163	210.0	35.8	17.0	11.9	5.7
	2165	464.3	49.7	10.7	16.7	3.6
	2166	759.0	56.9	7.5	19.0	2.5
S30	2257	319.0	51.5	16.1	17.2	5.4
	2258	583.3	64.0	11.0	21.3	3.7
	2259	998.6	64.0	6.4	21.3	2.1
	2260	1794.0	64.0	3.6	21.3	1.2
S40	2207	3660	120	3.2	40	1.1
	2208	4550	120	2.6	40	.8
	2256	1794	64	3.6	21.3	1.2
	2261	2340	64	2.7	21.3	.9

AS/400 Advanced Servers - V4R1 and V4R2

Model	Feature	Maximum Client/Serv CPW	Maximum Interactive CPW	CPU % at Maximum Interactive	1/3 Max Interactive CPW (Knee)	CPU % at the Knee
150	2269	20.2 c	13.8	51.1	4.6	10.0
	2269	27.0 u	13.8	51.1	4.6	10.0
	2270	20.2 c	20.2	61.9	6.7	10.0
	2270	35.0 u	20.6	61.9	6.9	10.0
40S	2109	27.0	9.4	30.1	3.1	10.0
	2110	35.0	14.5	37.4	3.9	10.0
	2111	63.0	21.6	29.8	7.2	10.0
	2112	91.0	32.2	29.8	10.8	10.0
50S	2120	81.6	22.5	27.8	8.1	10.0
	2121	111.5	32.2	30.0	10.7	10.0
	2122	138.0	32.2	23.8	12.0	8.9
53S	2154	188.2	32.2	20.3	15.9	10.0
	2155	319.0	32.2	13.5	10.7	4.5
	2156	598.0	32.2	9.0	10.7	3.0
	2157	650.0	32.2	7.7	10.9	2.6

Note: For the Model 150 c is constrained and u is unconstrained. A constrained CPW rating means the maximum memory or DASD configuration is the constraining factor, not the processor. An unconstrained CPW rating means the processor is the first constrained resource.

AS/400 Advanced Servers - V3R7

Model	Feature	Maximum Client/Serv CPW	Maximum Interactive CPW	CPU % at Maximum Interactive	1/3 Max Interactive CPW (Knee)	CPU % at the Knee
150	2269	10.9 c	10.9	51.1	3.6	10.0
	2269	27.0 u	13.8	51.1	4.6	10.0
	2270	10.9 c	10.9	61.9	3.6	10.0
	2270	33.3 u	20.6	61.9	6.9	10.0
40S	2109	27.0	9.4	30.1	3.1	10.0
	2110	33.3	13.8	37.4	3.7	10.0
	2111	59.8	20.6	29.8	6.9	10.0
	2112	87.3	30.7	29.8	10.3	10.0
50S	2120	77.7	21.4	27.8	7.7	10.0
	2121	104.2	30.7	30.0	10.2	10.0
	2122	130.7	30.7	23.8	11.5	8.9
53S	2154	162.7	30.7	20.3	15.1	10.0
	2155	278.8	30.7	13.5	10.2	4.5
	2156	459.3	30.7	9.0	10.2	3.0
	2157	509.9	30.7	7.7	10.4	2.6

Note: For the Model 150 c is constrained and u is unconstrained. A constrained CPW rating means the maximum memory or DASD configuration is the constraining factor, not the processor. An unconstrained CPW rating means the processor is the first constrained resource.

AS/400 Advanced Servers - V3R6

Model	Feature	Maximum Client/Serv CPW	Maximum Interactive CPW	CPU % at Maximum Interactive	1/3 Max Interactive CPW (Knee)	CPU % at the Knee
40S	2109	24.5	8.4	30.1	2.8	10.0
	2110	30.6	12.3	37.4	3.3	10.0
	2111	52.9	18.3	29.8	6.1	10.0
	2112	77.3	26.9	29.8	9.0	10.0
50S	2120	66.7	18.7	27.8	6.7	10.0
	2121	85.0	26.9	30.0	9.0	10.0
	2122	106.8	26.9	23.8	10.1	8.9
53S	2154	132.5	26.9	20.3	13.3	10.0
	2155	198.7	26.9	13.5	9.0	4.5
	2156	299.0	26.9	9.0	9.0	3.0
	2157	349.8	26.9	7.7	9.1	2.6

CPW values may be increased as enhancements are made to the operating system (e.g. each feature of the Model 53S for V3R6, V3R7 and V4R1). The server model behavior is fixed to the original CPW values. For example, a model 53S-2157 had V3R6 original CPWs of 349.8/26.9 and for V4R1 has CPWs 650.0/32.2. When using the 53S with V4R1, this means the knee of the curve is 2.6% CPU and the maximum interactive is 7.7% CPU, the same as it was in V3R6.

Custom Server Model Behavior

On a custom server model, there are two CPW values. The larger value represents the maximum workload the model could support if the workload were entirely client/server (i.e. no interactive components). The smaller CPW value represents the maximum workload the model could support if the workload were entirely interactive. **These values are NOT additive - interactive processing will reduce the system′s client/server processing capability.**

Custom server models interact to batch and interactive workloads similar to the server models but the degree of interaction and priority of workloads follows a different algorithm and hence the knee of the curve for workload interaction is at a different point. The custom servers offer excellent client/server workload capability and much higher interactive workload capability compared to the standard server models.

Custom server models were designed for a client/server workload and to accommodate a fairly high interactive workload. When the interactive workload exceeds approximately 6/7 of the maximum interactive CPW, the client/server processing performance of the system becomes increasingly impacted. Once the interactive workload reaches the maximum interactive CPW value, all the CPU cycles are being used and there is no capacity available for handling client/server tasks.

The following figure graphically represents the above for a model S20-2178. For this model, note that below about 25% (the ″knee of the curve″), the effective interactive utilization is roughly equal to the useable interactive utilization. Also note the substantially increased rate of total CPU utilization (due to interactive workload) once the useable interactive workload exceeds 25%. Finally, note that the CPU is completely utilized once the useable interactive workload reaches about 29.2%. At this saturation point, there is no CPU available for client/server.

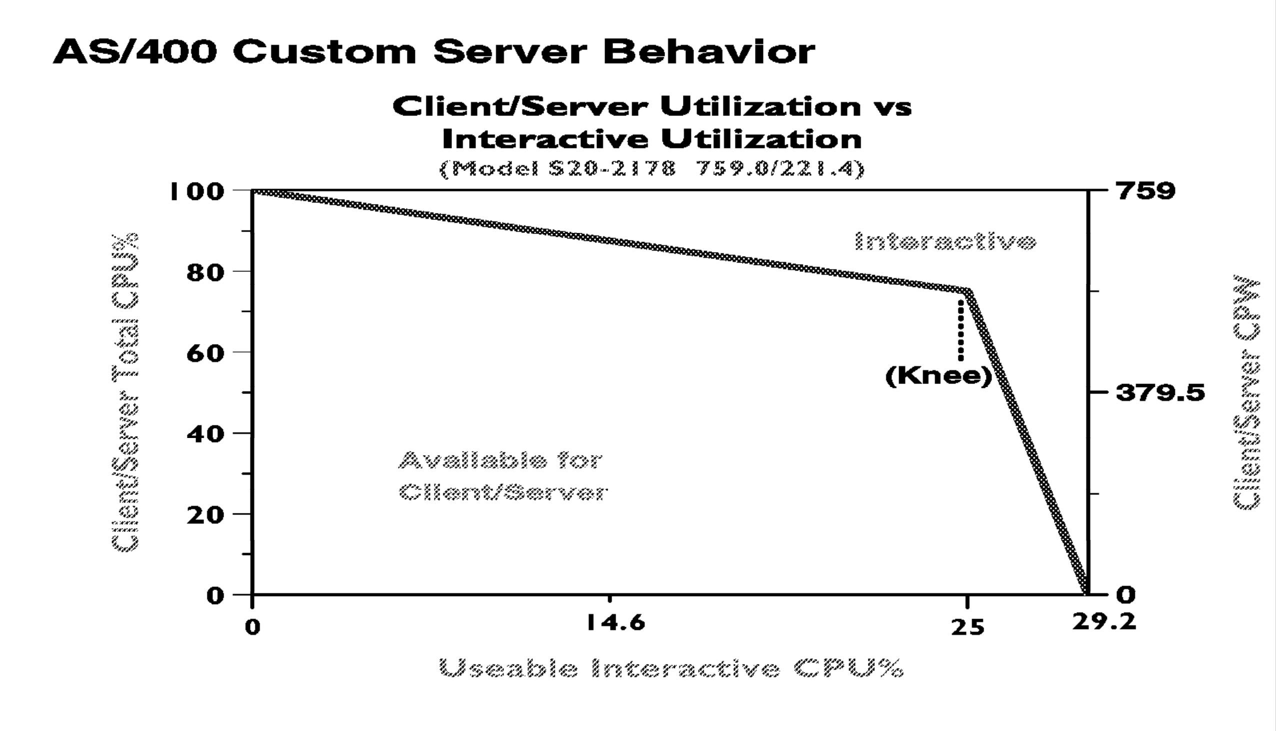


Figure 7.

Similar figures may be constructed for other server models by simply using the applicable values from the tables below for the knee of the curve and for the maximum interactive CPU.

Customers should plan for their interactive workloads to execute below (less than) the knee of the curve.

In general, executing a single interactive job will not cause a significant impact to client/server performance.

The figure above shows a straight line for the effective interactive utilization, which is the intent of the server behavior. Real/customer environments will produce a curved line since most environments will be dynamic, due to job initiation, interrupts, etc, etc.

The following table shows, for each custom server model, the maximum interactive CPW and its corresponding CPU % and the point (about 6/7 maximum interactive - the knee of the curve) where the interactive utilization begins to increasingly impact client/server performance.

The knee of the curve is given in CPW values as well as CPU %. For those models that are n-ways (multiple processors), the percent value is the percent of all the processors (not of a single one).

AS/400e custom servers - V4R1 and V4R2

		Maximum	Maximum	CPU % at	6/7 Max	CPU % at
Model	Feature	Client/Serv	Interactive	Maximum	Interactive	the
		CPW	CPW	Interactive	CPW (Knee)	Knee
S20	2177	759.0	110.7	14.6	94.9	12.5
	2178	759.0	221.4	29.2	189.8	25.0
S30	2320	998.6	215.1	21.5	184.4	18.5
	2321	1794.0	386.4	21.5	331.2	18.5
	2322	1794.0	579.6	32.3	496.8	27.7

AS/400e custom servers - V4R3

		Maximum	Maximum	CPU % at	6/7 Max	CPU % at
Model	Feature	Client/Serv	Interactive	Maximum	Interactive	the
		CPW	CPW	Interactive	CPW (Knee)	Knee
S40	2340	3660	1050	28.6	900	24.5
	2341	4550	2050	38.6	1757.1	33.1

The CPW values shown in the tables above are based on IBM internal tests. Actual performance in a customer environment will vary.

Server and Custom Server Model Behavior

If interactive CPW utilization is above the knee of the curve, the following calculations can be used to determine the effective interactive utilization or the available/remaining client/server CPW. These equations apply to all server models.

```
EIU             = Effective interactive utilization
CPWcs(maximum)  = client/server CPW maximum value
CPWint(maximum) = interactive CPW maximum value
CPWint(knee)    = interactive CPW at the knee of the curve
CPWint(Y)       = interactive CPW of the desired workload

X   = (CPWint(Y) - CPWint(knee)) / (CPWint(maximum) - CPWint(knee))
EIU = CPWint(knee) + (X * (CPWcs(maximum) - CPWint(knee)))
CPW remaining for batch = CPWcs(maximum) - EIU
```

For example, a S20-2178 has CPW values of 759/221.4. Assume you want an interactive CPW of 200.

```
CPWcs(maximum)  = 759
CPWint(maximum) = 221.4
CPWint(knee)    = 25
CPWint(Y)       = 200

X = (200 - 25) / (221.4 - 25)
  = .891

EIU = 25 + (.891 * (759 - 25))
    = 678.99

CPW remaining for batch = 759 - 678.9
                        = 80.01
```

Performance highlights of new Model 170 servers

The new Model 170 servers (features 2290, 2291, 2292, 2385 and 2386) are significantly more powerful and deal with additional interactive workload in a more straight forward manner than the previous Model 170′s announced in Feb ′97. The new Model 170s have a faster processor (262MHz vs 125MHz) and more main memory (up to 3.5GB vs 1.0GB). In addition, the interactive workload balancing algorithm has been improved to provide a linear relationship between the published client/server (batch) and interactive workloads as measured by CPW.

The CPW rating for the maximum Client/Server workload now reflects the relative processor capacity rather than the ″system capacity″ and therefore there is no need to state a ″constrained performance″ CPW.

As interactive workload is added to the Model 170 systems, the remaining available client/server (batch) workload available is calculated as:

CPW (C/S batch) = CPW(processor) - CPW(interactive).

This is valid up to the published Interactive CPW rating. As long as the interactive CPW workload does not exceed the published interactive value, then interactive performance and client/server (batch) workloads will be both be optimized for best performance. Bottom line, customers can use the entire interactive capacity with no impacts to client/server (batch) workload response times.

Relative performance measurements are derived from commercial processing workload (CPW) on AS/400. CPW is representative of commercial applications, particularly those that do significant database processing in conjunction with journaling and commitment control.

The new Model 170s have new performance definitions:

- **Processor Capacity** (CPW) represents the relative performance (maximum capacity) of a processor feature running a commercial processing workload (CPW) in a client/server environment. This value can be used to compare relative performance between processor features.

  Processor capacity is achievable when the commercial workload is not constrained by main storage or DASD. This value was stated as ″unconstrained C/S CPW″ for the Model 170 in prior releases.

- **Interactive Capacity** (CPW) represents the relative performance available to perform host-centric workloads. The amount of interactive capacity consumed will reduce the available processor capacity by the same amount.

On the existing Model 170′s the server algorithm kicked in at 1/3 of the Interactive CPW and very quickly added CFINT workload to control the available processor power for C/S (batch) workloads. With the new models, this overhead does not occur until the published interactive capacity is reached. This allows the new models to operate more smoothly in interactive and mixed environments. If or when, interactive workload exceeds the published limit, then CFINT workload is added and client/server (batch) capacity will be quickly reduced and become zero when interactive hits 7/6 of the published interactive CPW for a given model.

The absolute limit for dedicated interactive capacity on the new models can be computed by multiplying the published interactive CPW rating by a factor of 7/6. The absolute limit for dedicated client/server (batch) is the published processor capacity value. This assumes that sufficient disk and memory as well as other system resources are available to fit the needs of the customer′s programs, etc. Customer workloads that would require more than 10 disk arms for optimium performance should not be expected to give optimium performance on the Model 170, as 10 disk access arms are the maximum configuration.

When the new Model 170 servers are running less than the published interactive workload, no Server Dynamic Tuning (SDT) is necessary to achieve balanced performance between interactive and client/server (batch) workloads. However, as with previous mixed-mode and server models, a system value (QYNPTYADJ - Server Dynamic Tuning (SDT)) is available to determine how the server will react to work requests when interactive workload exceeds the ″knee″ (point where overhead is introduced). If the server (SDT) value is turned on, non-interactive work is favored over additional interactive work. If it is turned off, additional interactive work is allowed at the expense of low-priority non-interactive work. With existing model 170 servers, this value affected the server when interactive requirements exceeded 1/3 of the interactive capacity of the server. With the new Model 170 servers, this value only affects the server when interactive requirements exceed the published interactive capacity rating.

New Model 170s

	Processor	Interactive	Interactive	CPU % by	CPU % by	CPU % by
Feature	CPW	CPW	CPW	Interact	Processor	Interact
		(knee)	(max)	at Knee	at Knee	at Max
2290	73	20	23.3	26.6	73.4	31.1
2291	115	25	29.1	22.7	77.3	26.5
2292	220	30	35	14.4	85.6	16.9
2385	460	50	58.3	10.3	89.7	12
2386	460	70	81.6	15.5	84.5	18.1

Note: the CPU not used by the interactive workloads at their Max CPU is used by the system CFINTnn jobs. For example, for the 2386 model the interactive workloads use 18.1% of the CPU at their maximum and the CFINTnn jobs use the remaining 71.9%. The processor workloads use 0% CPU when the interactive workloads are using their maximum value.

Actual customer results vary depending on specific workload characteristics. Because of the faster processor speed of the new Model 170 servers, most performance improvements will be seen in CPU intensive workloads.

Existing model 170s

	C	Client	Interactive	Interactive	CPU % by	CPU % by
Feature	or	Server	CPW	CPW	Interact	Interact
	U	CPW	(max)	(knee)	at Knee	at Max
2159	c	73	16	5.3	7.7	22.2
2159	u	73	16	5.3	7.7	22.2
2160	c	114	23	7.7	7.4	21.2
2160	u	114	23	7.7	7.4	21.2
2164	c	125	29	9.7	4.7	14
2164	u	210	29	9.7	4.7	14
2176	c	125	40	13.3	4.4	12.9
2176	u	319	40	13.3	4.4	12.9
2183	c	125	67	22.3	7.2	21.5
2183	u	319	67	22.3	7.2	21.5

Note: A constrained (c) CPW rating means the maximum memory or DASD configuration is the constraining factor, not the processor. An unconstrained (u) CPW rating means the processor is the first constrained resource.

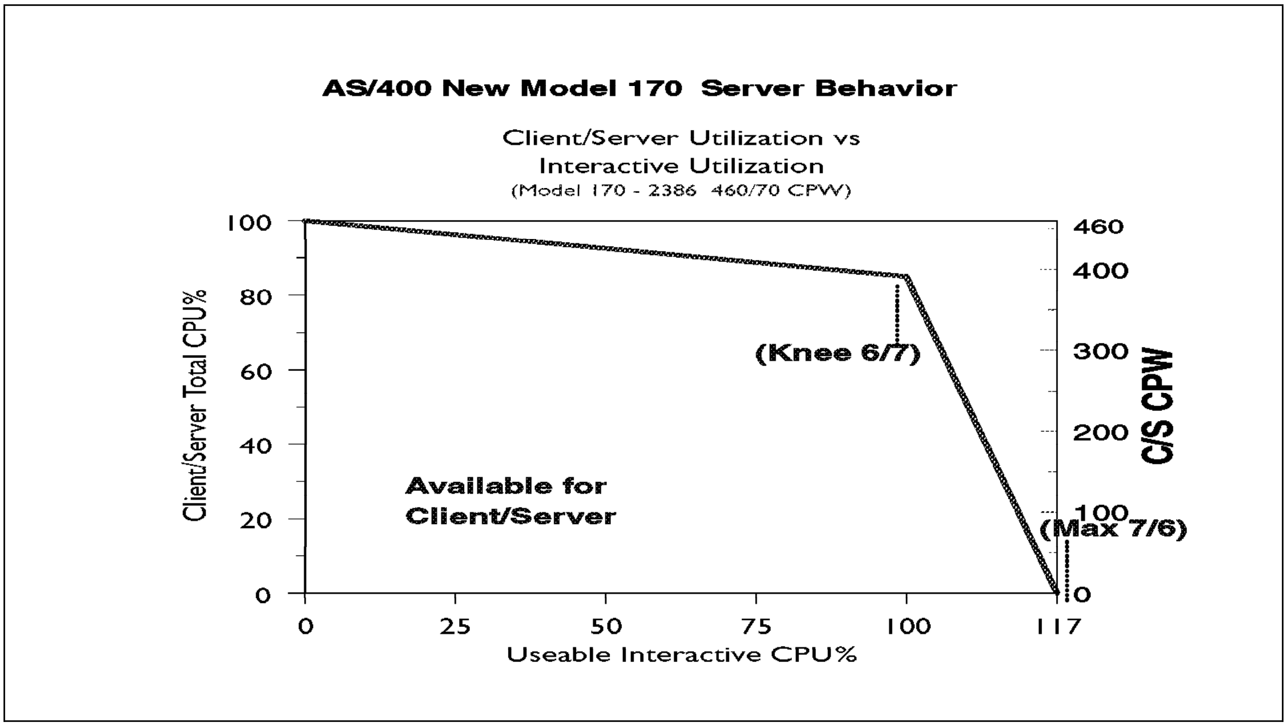
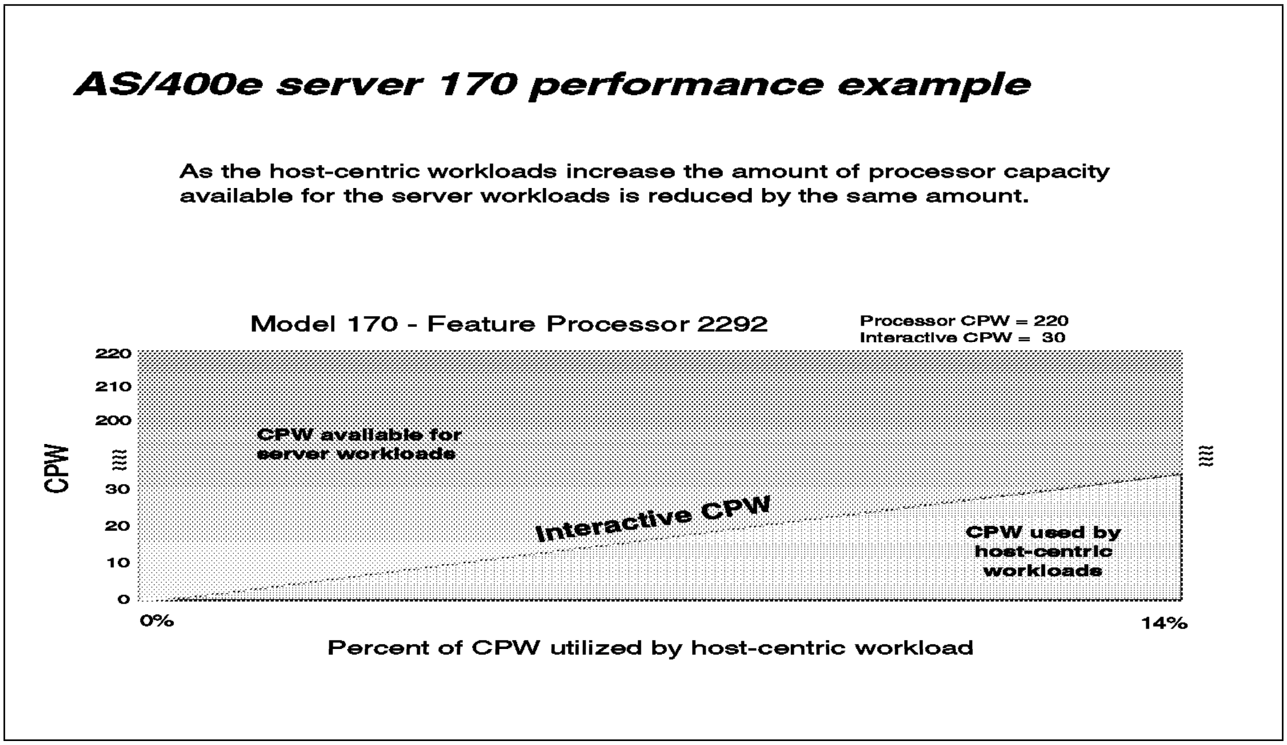


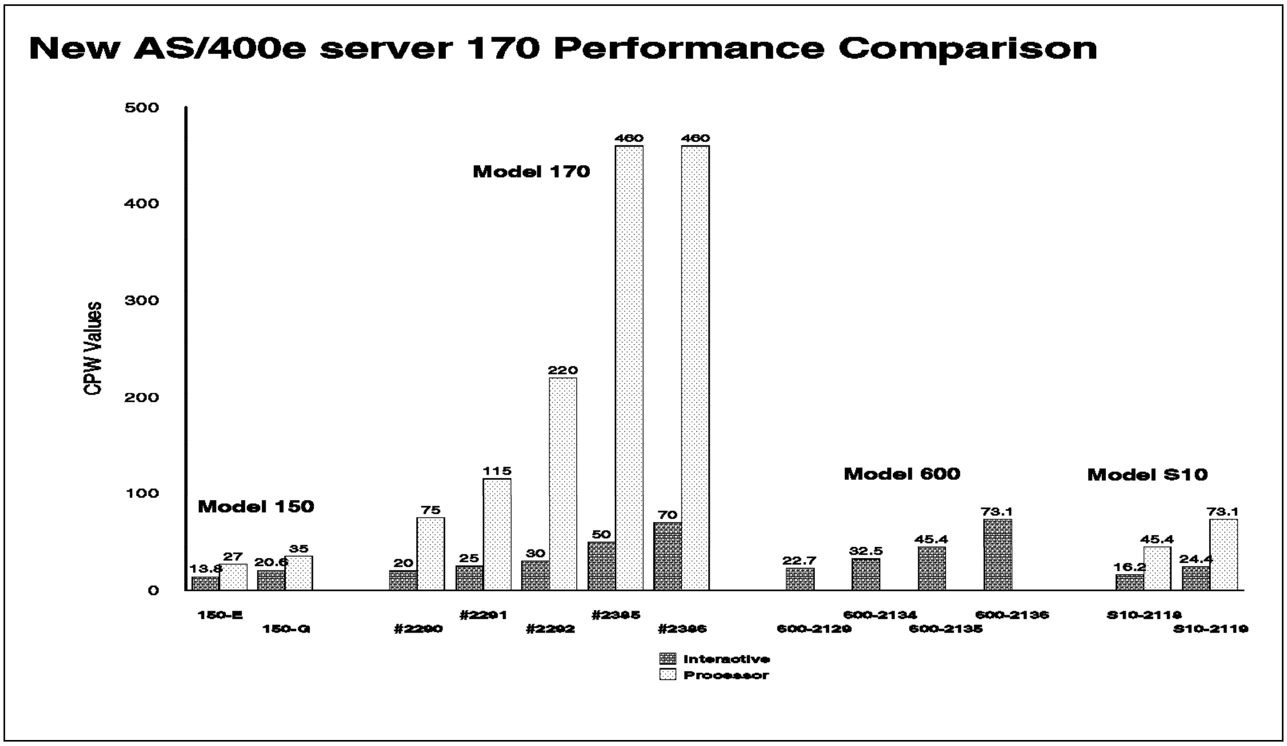
Figure 8.



AS/400e server 170 performance example

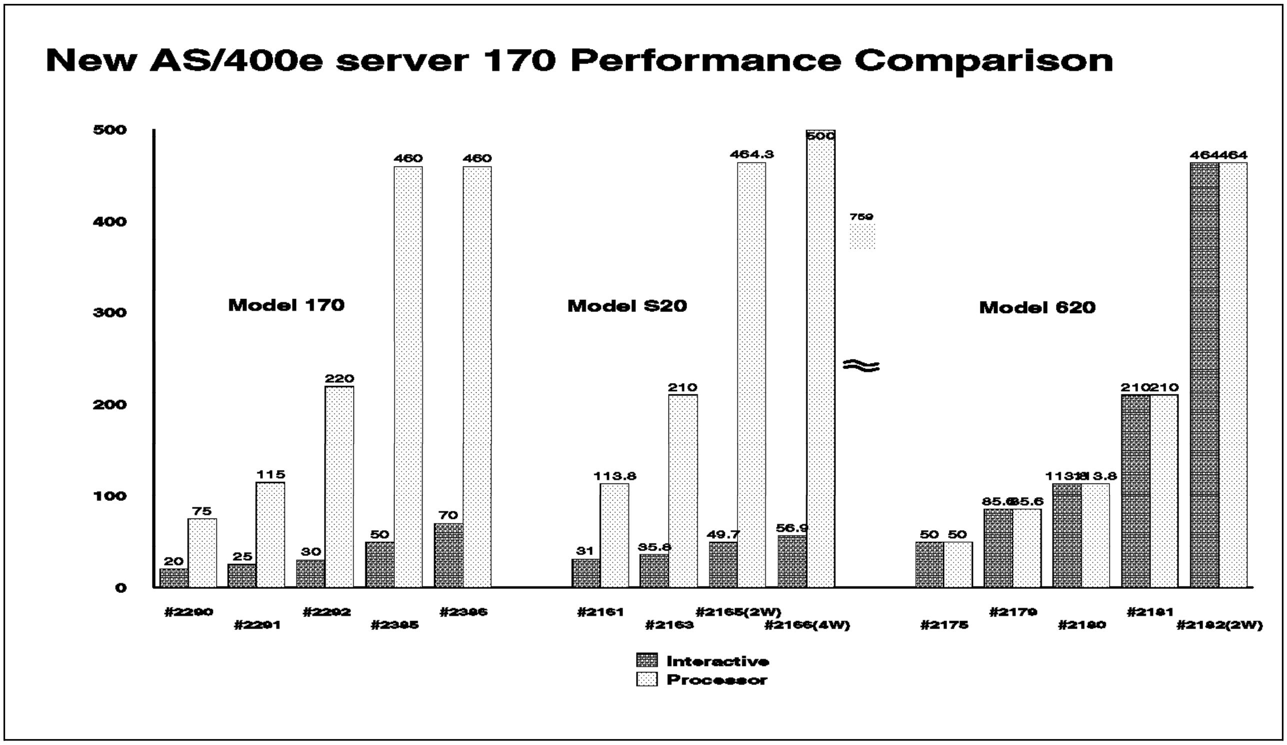
As the host-centric workloads increase the amount of processor capacity available for the server workloads is reduced by the same amount.

Figure 9.



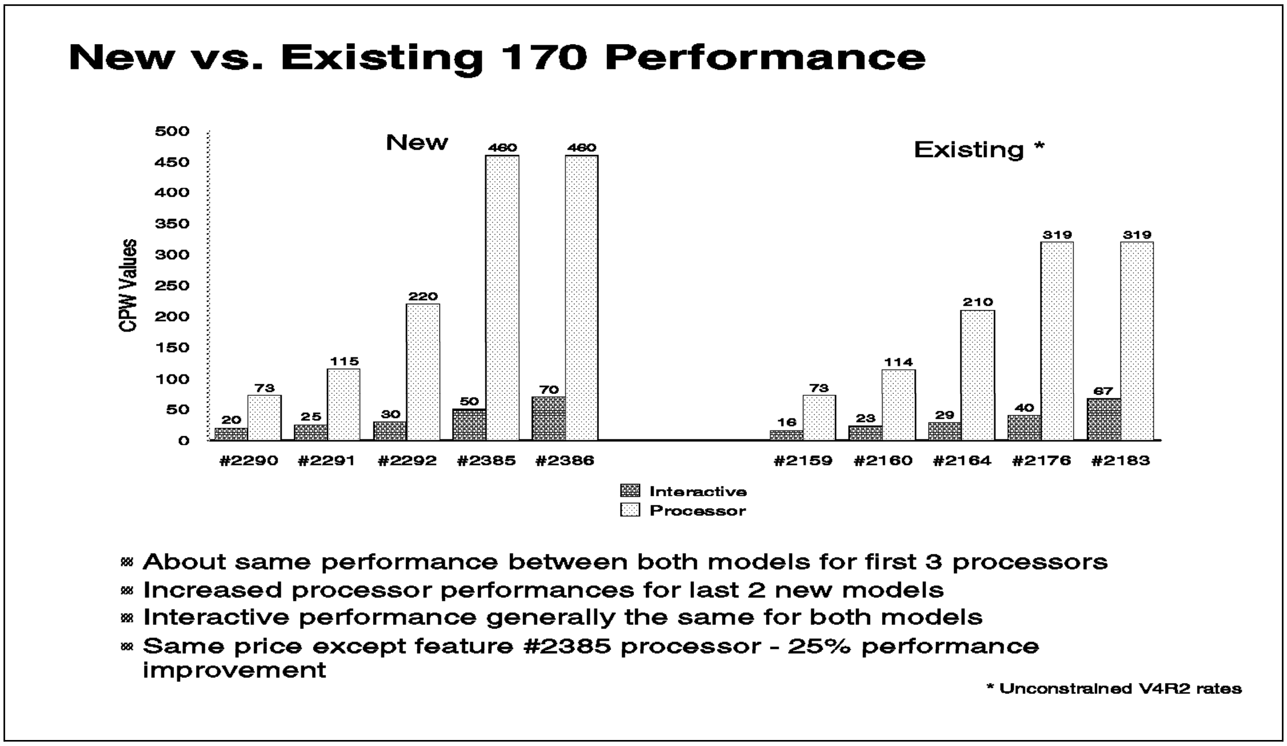
New AS/400e server 170 Performance Comparison

Figure 10.



New AS/400e server 170 Performance Comparison

Figure 11.



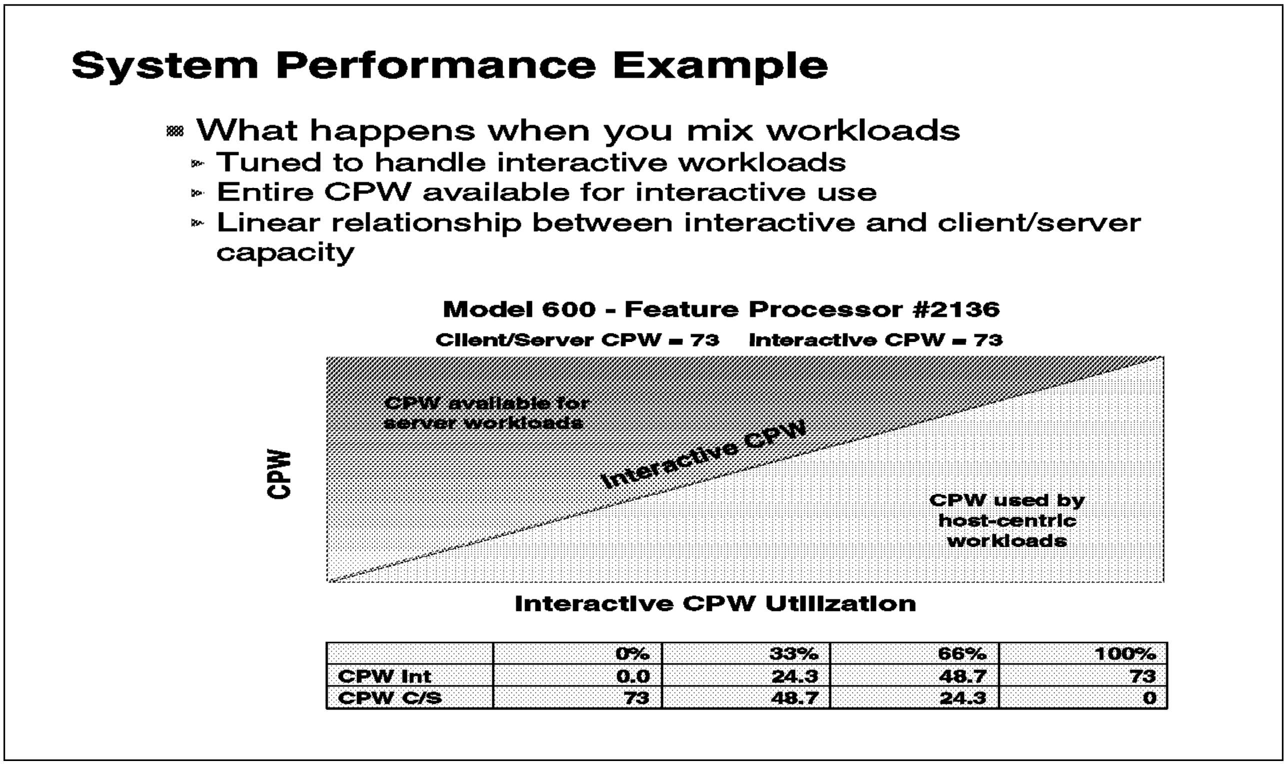
New vs. Existing 170 Performance

- About same performance between both models for first 3 processors
- Increased processor performances for last 2 new models
- Interactive performance generally the same for both models
- Same price except feature #2385 processor - 25% performance improvement

* Unconstrained V4R2 rates

Figure 12.

System, Server, Custom Server, and S170 Performance Examples



System Performance Example

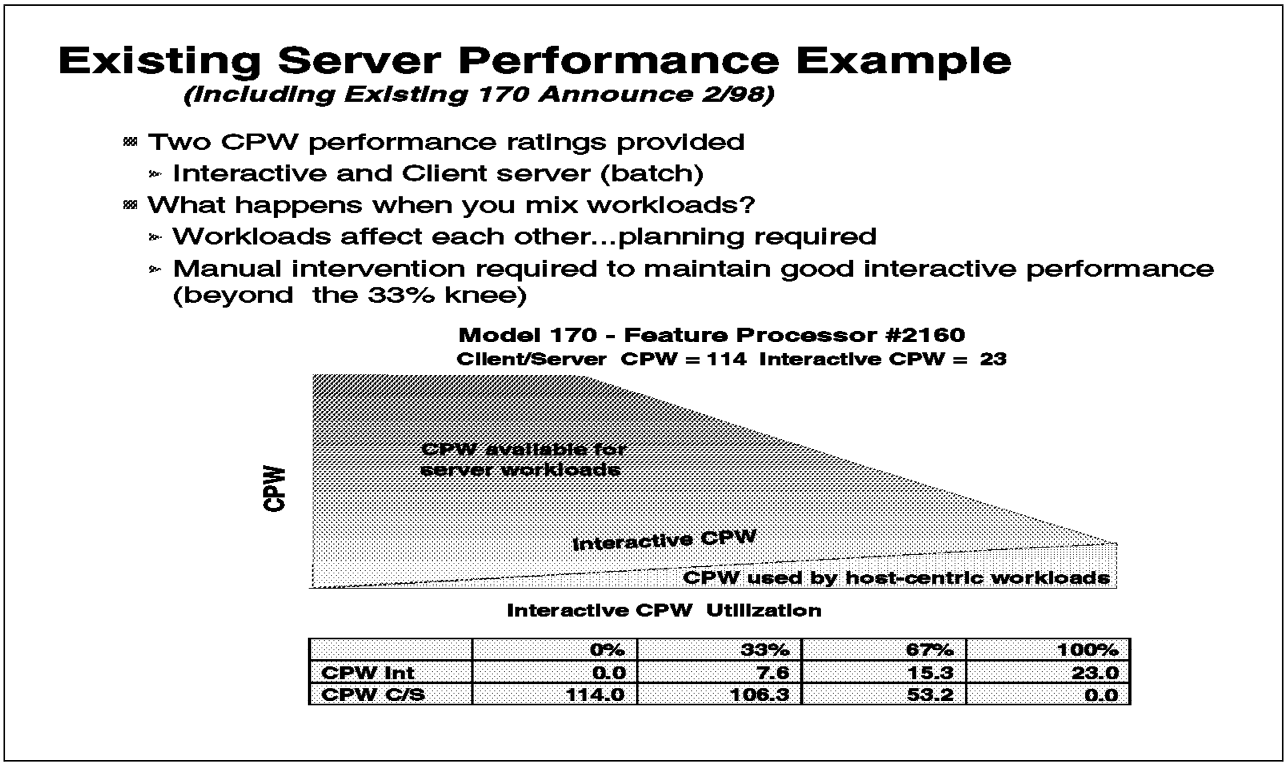
- What happens when you mix workloads
  - Tuned to handle interactive workloads
  - Entire CPW available for interactive use
  - Linear relationship between interactive and client/server capacity

Model 600 - Feature Processor #2136

Client/Server CPW = 73 Interactive CPW = 73

	0%	33%	66%	100%
CPW Int	0.0	24.3	48.7	73
CPW C/S	73	48.7	24.3	0

Figure 13.



Existing Server Performance Example

(Including Existing 170 Announce 2/98)

- Two CPW performance ratings provided
  - Interactive and Client server (batch)
- What happens when you mix workloads?
  - Workloads affect each other...planning required
  - Manual intervention required to maintain good interactive performance (beyond the 33% knee)

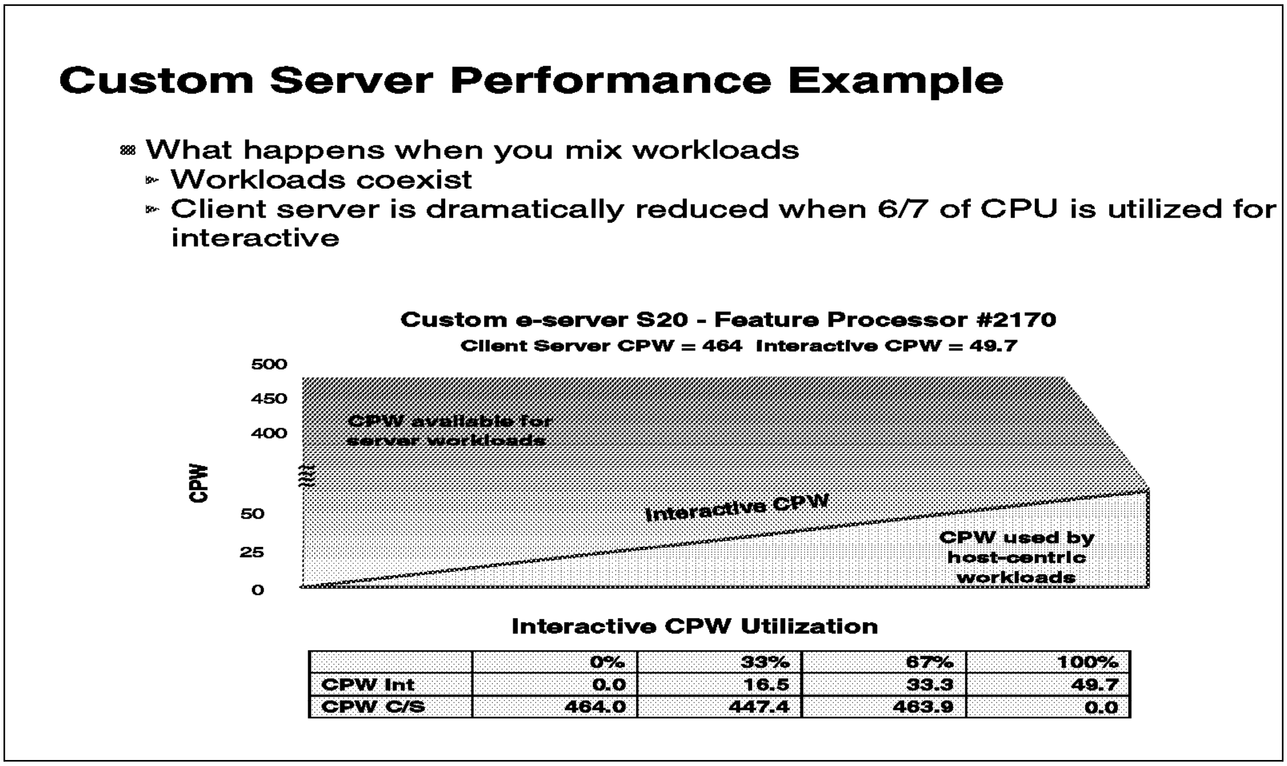
Model 170 - Feature Processor #2160

Client/Server CPW = 114 Interactive CPW = 23

CPW available for server workloads
Interactive CPW
CPW used by host-centric workloads
CPW
Interactive CPW Utilization

	0%	33%	67%	100%
CPW Int	0.0	7.6	15.3	23.0
CPW C/S	114.0	106.3	53.2	0.0

Figure 14.



Custom Server Performance Example

- What happens when you mix workloads
  - Workloads coexist
  - Client server is dramatically reduced when 6/7 of CPU is utilized for interactive

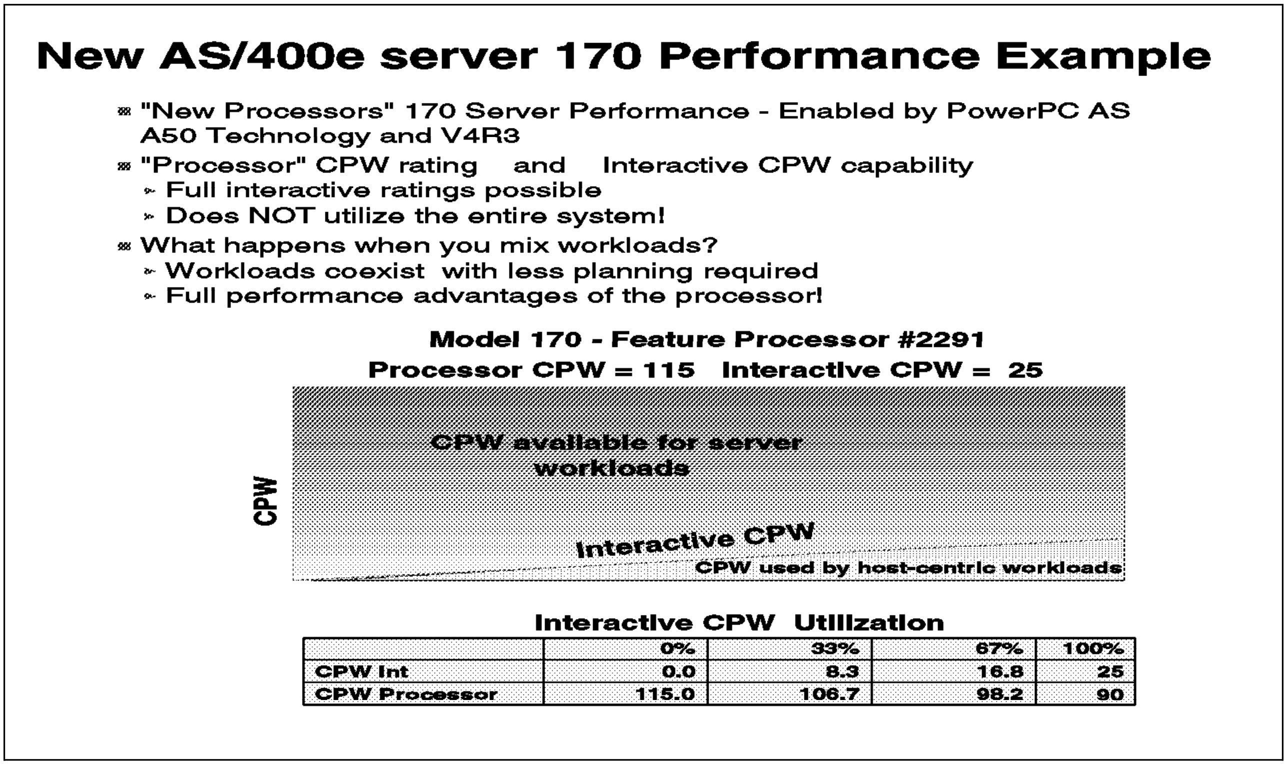
Custom e-server S20 - Feature Processor #2170

Client Server CPW = 464 Interactive CPW = 49.7

Interactive CPW Utilization

	0%	33%	67%	100%
CPW Int	0.0	16.5	33.3	49.7
CPW C/S	464.0	447.4	463.9	0.0

Figure 15.



New AS/400e server 170 Performance Example

- "New Processors" 170 Server Performance - Enabled by PowerPC AS A50 Technology and V4R3
- "Processor" CPW rating and Interactive CPW capability
  - Full interactive ratings possible
  - Does NOT utilize the entire system!
- What happens when you mix workloads?
  - Workloads coexist with less planning required
  - Full performance advantages of the processor!

Model 170 - Feature Processor #2291

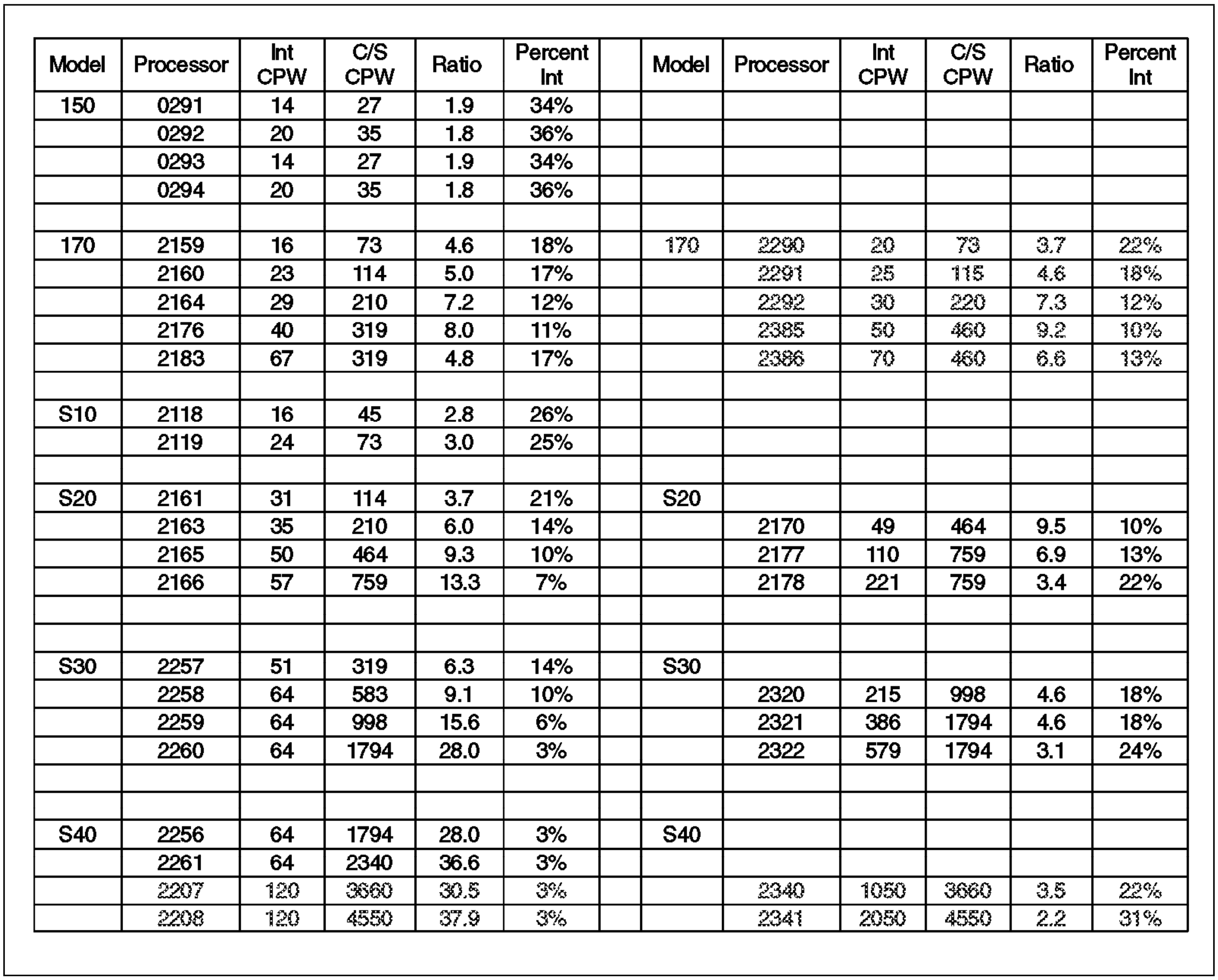
Processor CPW = 115 Interactive CPW = 25

CPW
CPW available for server workloads
Interactive CPW
CPW used by host-centric workloads

Interactive CPW Utilization

	0%	33%	67%	100%
CPW Int	0.0	8.3	16.8	25
CPW Processor	115.0	106.7	98.2	90

Figure 16.



Model	Processor	Int CPW	C/S CPW	Ratio	Percent Int		Model	Processor	Int CPW	C/S CPW	Ratio	Percent Int
150	0291	14	27	1.9	34%							
	0292	20	35	1.8	36%							
	0293	14	27	1.9	34%							
	0294	20	35	1.8	36%							
170	2159	16	73	4.6	18%		170	2290	20	73	3.7	22%
	2160	23	114	5.0	17%			2291	25	115	4.6	18%
	2164	29	210	7.2	12%			2292	30	220	7.3	12%
	2176	40	319	8.0	11%			2385	50	460	9.2	10%
	2183	67	319	4.8	17%			2386	70	460	6.6	13%
S10	2118	16	45	2.8	26%							
	2119	24	73	3.0	25%							
S20	2161	31	114	3.7	21%		S20					
	2163	35	210	6.0	14%			2170	49	464	9.5	10%
	2165	50	464	9.3	10%			2177	110	759	6.9	13%
	2166	57	759	13.3	7%			2178	221	759	3.4	22%
S30	2257	51	319	6.3	14%		S30					
	2258	64	583	9.1	10%			2320	215	998	4.6	18%
	2259	64	998	15.6	6%			2321	386	1794	4.6	18%
	2260	64	1794	28.0	3%			2322	579	1794	3.1	24%
S40	2256	64	1794	28.0	3%		S40					
	2261	64	2340	36.6	3%							
	2207	120	3660	30.5	3%			2340	1050	3660	3.5	22%
	2208	120	4550	37.9	3%			2341	2050	4550	2.2	31%

Figure 17.

Note: numbers are rounded.

Ratio = C/S CPW / Int CPW

Percent Int = Int CPW / (Int CPW + C/S CPW)

For example, for the model S40, model 2207:

```
Ratio = 3660 / 120
      = 30.5

Percent Int = 120 / (120 + 3660)
            = 3 %
```

Server Dynamic Tuning (SDT)

Logic was added in V4R1 and is still in use today so customers could better control the impact of interactive work on their client/server performance. Note that with the new Model 170 servers (features 2290, 2291, 2292, 2385 and 2386) this logic only affects the server when interactive requirements exceed the published interactive capacity rating. For futher details see "Performance highlights of new Model 170 servers" on page 22.

Through dynamic prioritization, all interactive jobs will be put lower in the priority queue, approximately at the knee of the curve. Placing the interactive jobs at a lesser priority causes the interactive jobs to slow down, and more processing power to be allocated to the client/server processing. As the interactive jobs receive less processing time, their impact on client/server processing will be lessened. When the interactive jobs are no longer impacting the client/server jobs, their priority will dynamically be raised again.

The dynamic prioritization acts as a regulator which can help reduce the impact to client/server processing when additional interactive workload is placed on the system. In most cases, this results in better overall throughput when operating in a mixed client/server and interactive environment, but it can cause a noticeable slowdown in interactive response.

To fully enable SDT, customers MUST use a NON-INTERACTIVE job run priority (RUNPTY parameter) value of 35 or less (which raises the priority, closer to the default priority of 20 for interactive jobs).

Changing the existing non-interactive job run priority can be done either through the Change Job (CHGJOB) command or by changing the RUNPTY value of the Class Description object used by the non-interactive job. This includes IBM-supplied or application provided class descriptions.

Examples of IBM-supplied class descriptions with a run priority value higher than 35 include QBATCH and QSNADS and QSYSCLS50. Customers should consider changing the RUNPTY value for QBATCH and QSNADS class descriptions or changing subsystem routing entries to not use class descriptions QBATCH, QSNADS, or QSYSCLS50.

If customers modify an IBM-supplied class description, they are responsible for ensuring the priority value is 35 or less after each new release or cumulative PTF package has been installed. One way to do this is to include the Change Class (CHGCLS) command in the system Start Up program.

NOTE: Several IBM-supplied class descriptions already have RUNPTY values of 35 or less. In these cases no user action is required. One example of this is class description QPWFSERVER with RUNPTY(20). This class description is used by Client Access database server jobs QZDAINIT (APPC) and QZDASOINIT (TCP/IP).

The system deprioritizes jobs according to groups or ″bands″ of RUNPTY values. For example, 10-16 is band 1, 17-22 is band 2, 23-35 is band 3, and so on.

Interactive jobs with priorities 10-16 are an exception case with V4R1. Their priorities will not be adjusted by SDT. These jobs will always run at their specified 10-16 priority.

When only a single interactive job is running, it will not be dynamically reprioritized.

When the interactive workload exceeds the knee of the curve, the priority of all interactive jobs is decreased one priority band, as defined by the Dynamic Priority Scheduler, every 15 seconds. If needed, the priority will be decreased to the 52-89 band. Then, if/when the interactive CPW work load falls below the knee, each interactive job′s priority will gradually be reset to its starting value (one priority band every 15 seconds), when the job is dispatched.

If the priority of non-interactive jobs are not set to 35 or lower, SDT stills works, but its effectiveness is greatly reduced, resulting in server behavior more like V3R6 and V3R7. That is, once the knee is exceeded, interactive priority is automatically decreased. Assuming non-interactive is set at priority 50, interactive could eventually get decreased to the 52-89 priority band. At this point, the processor is slowed and interactive and non-interactive are running at about the same priority. (There is little priority difference between 47-51 band and the 52-89 band.)

If the Dynamic Priority Scheduler is turned off, SDT is also turned off.

Note that even with SDT, the underlying server behavior is unchanged. Customers get no more CPU cycles for either interactive or non-interactive jobs. SDT simply tries to regulate interactive jobs once they exceed the knee of the curve.

Obviously systems can still easily exceed the knee and stay above it, by having a large number of interactive jobs, by setting the priority of interactive jobs in the 10-16 range, by having a small client/server workload with a modest interactive workload, etc. The entire server behavior is a partnership with customers to give non-interactive jobs the bulk of the CPU while not entirely shutting out interactive.

Migration from Traditional Models

This section describes a suggested methodology to determine which server model is appropriate in a move from a traditional model. It is assumed that the server model will have both interactive and client/server workloads. This section primarily describes how to determine which server model can contain the interactive workload of the current traditional model when migration of a workload is occurring.

To get the same performance and response time, from a CPU perspective, the interactive CPU utilization of the current traditional model must be known. Traditional CPU utilization can be determined in a number of ways. One way is to sum up the CPU utilization for interactive jobs shown on the Work with Active Jobs (WRKACTJOB) command. See the following WRKACTJOB screen example:

```
**********************************************************************
                          Work with Active Jobs
                                                           10/22/97
CPU%:   33.0   Elapsed time:   00:00:00   Active jobs:   152

Type options, press Enter.

 2=Change  3=Hold  4=End   5=Work with  6=Release  7=Display message
 8=Work with spooled files  13=Disconnect ...

Opt  Subsystem/Job  User       Type  CPU %  Function         Status
     QBATCH         QYSS       SBS      .0                    DEQW
     QCMN           QSYS       SBS      .0                    DEQW
     QCTL           QSYS       SBS      .0                    DEQW
       QSYSSCD      QPGMR      BCH      .0  PGM-QEZSCNEP      EVTW
     QINTER         QSYS       SBS      .0                    DEQW
       DSP05        TESTER     INT     0.2  PGM-BUPMENUNE     DSPW
       QPADEV0021   TEST01     INT      .7  CMD-WRKACTJOB     RUN
     QSERVER        QSYS       SBS      .0                    DEQW
       QPWFSERVSD   QUSER      BCH      .0                    SELW
       QPWFSERVS0   QUSER      PJ       .0                    DEQW
**********************************************************************
```

(Calculate the average of the CPU utilization for all job types ″INT″ for the desired time interval for interactive CPU utilization - ″P″ in the formula below.)

Another method is to run the Performance Monitor (STRPFRMON, ENDPFRMON) during selected time periods and review the first page of the Performance Tools/400 licensed program Component Report. The following is an example of this section of the report:

```
**********************************************************************
                                              Component Report
                                         Component Interval Activity
                                         Data collected 190396 at 1030
Member . . . : Q960791030 Model/Serial . : 310-2043/10-0751D Main st...
  Library. . : PFRRES     System name. . :          TEST01 Version/Re..
                                                             --Disk I/O --
 ITV       Tns        Rsp      ---- CPU Utilization ---  -- Per Second-
 End      /Hour      /Tns      Total     Inter    Batch  Sync     Async
-----    -------    ------     -----     -----    -----  --------------
10:36      6,164       .80      85.2      32.2     46.3   102.9    39.0
10:41      7,404       .94      91.3      45.2     39.5   103.3    33.9
10:46      5,466       .72      97.6      38.8     51.0    96.6    33.2
10:51      5,622      1.21      97.9      35.6     57.4    86.6    49.0
10:56      4,527       .76      97.9      16.5     77.4    64.2    40.7
11:01      4,912       .79      95.9      25.0     68.8    73.1    32.3
  :          :         :          :         :        :       :       :
  :          :         :          :         :        :       :       :
11:46      6,528       .97      99.6      29.8     66.1    56.4    37.9
11:51      5,068      1.77      99.9      74.2     25.7    56.5    19.9
11:56      5,991      2.36      99.9      46.8     45.5    65.5    32.6

Itv End              -- Interval end time (hour and minute)
Tns /Hour            -- Number of interactive transactions per hour
Rsp /Tns             -- Average interactive transaction response time
  :          :         :          :         :        :       :       :
  :          :         :          :         :        :       :       :
**********************************************************************
```

(Calculate the average of the CPU utilization under the ″Inter″ heading for the desired time interval for interactive CPU utilization - ″P″ in the formula below.)

It is possible to have interactive jobs that do not show up with type ″INT″ or in the Performance Monitor Component Report. An example is a job that is submitted as a batch job that acquires a work station. These jobs should be included in the interactive CPU utilization count.

Most systems have peak workload environments. Care must be taken to insure that peaks can be contained in server model environments. **Some environments could have peak work loads that exceed the interactive capacity of a server model or could cause unacceptable response times and throughput.**

In the following equations, let the interactive CPU utilization of the existing traditional system be represented by percent P. A server model that should then produce the same response time and throughput would have a CPW of:

```
Server interactive CPW = 3 * P * Traditional CPW
```

or for custom models use:

```
Server interactive CPW = 1.0 * P * Traditional CPW (when P is less than 85%)
or
Server interactive CPW = 1.5 * P * Traditional CPW (when P greater than 85%)
```

You use the 1.5 factor to ensure the custom server is sized less than 85% CPU utilization.

These equations provide the server interactive CPU cycles required to keep the interactive utilization at or below the knee of the curve, with the current interactive workload. The equations given at the end of the **Server and Custom Server Model Behavior** section can be used to determine the effective interactive utilization above the knee of the curve. The interactive workload below the knee of the curve represents one third of the total possible interactive workload, for non-custom models. The equation shown in this section will migrate a traditional system to a server system and keep the interactive workload at or below the knee of the curve, that is, using less than two-thirds of the total possible interactive workload. In some environments these equations will be too conservative. A value of 1.2, rather than 1.5 would be less conservative. The equations presented in the Server and Custom Server Behavior section should be used by those customers who understand how server models work above the knee of the curve and the ramifications of the V4R1 enhancement.

These equations are for migration of ″existing workload″ situations only. Workload projections for ″initial installation″ of new custom servers are generally sized by the business partner for 50 - 60% CPW workloads and no ″formula increase″ would be needed.

For example, assume a model 510-2143 with a single V3R6 CPW rating of 66.7 and assume the Performance Tools/400 report lists interactive work CPU utilization as 21%. Using the previous formula, the server model must have an interactive CPW rating of at least 42 to maintain the same performance as the 510-2143.

```
  Server interactive CPW = 3 * P * Traditional CPW
                         = 3 * .21 * 66.7
                         = 42
```

A server model with an interactive CPW rating of at least 42 could approximate the same interactive work of the 510-2143, and still leave system capacity available for client/server activity. An S20-2165 is the first AS/400e series with an acceptable CPW rating (49.7).

Note that interactive and client/server CPWs are not additive. Interactive workloads which exceed (even briefly) the knee of the curve will consume a disproportionate share of the processing power and may result in insufficient system capacity for client/server activity and/or a significant increase in interactive response times.

Migration from Server Models

The section describes a recommended methodology for migrating from a server to a traditional model.

First determine the interactive CPU utilization for the server model. The second step is to determine the batch (client/server) CPU utilization for the server model. The previous section (″Migration from Traditional Models″) describes how the Work with Active Jobs (WRKACTJOB) command or the Performance Monitor (STRPFRMON, ENDPFRMON) may be used to gather this information. The last step is to get the Maximum Client/Server CPW rating for the server.

Now in the following equations, let:

I = Interactive CPU Utilization

B = Batch CPU Utilization

CPWcs = Maximum Client/Server CPW rating

A traditional model that should then produce the same response time and throughput would have a CPW of:

Traditional CPW = CPWcs * (I + B) / .70

Note: In the above formula the division by 70 percent (.70) is done as a guideline to keep the system′s CPU utilization at 70 percent, or less.

For example, assume a model 170-2160 with a V4R2 Maximum Client/Server CPW rating of 114, and assume the Performance Tools/400 report lists interactive work CPU utilization as 10% and batch CPU utiliziation at 50%. Using the previous formula, the traditional model should have a CPW rating of at least 97.7 to maintain the same performance as the 170-2160, this corresponds to an AS/400e 620-2180 system.

```
  Traditional CPW = CPWcs * (I + B) / .70
                  = 114 * (.10 + .50) / .70
                  = 97.7
```

This formula should ensure that this system will give similar performance, however; each situation is unique and should be evaluated with an understanding of what the performance goals are. For example, if longer batch execution times are acceptable then a system with a lower CPW rating may be sufficient.

Chapter 3. Batch Performance

In a commercial environment, batch workloads tend to be I/O intensive rather than CPU intensive. The factors that affect batch throughput for a given batch application include the following:

- Memory (Pool size)
- CPU (processor speed)
- DASD (number and type)
- System tuning parameters

Batch Workload Description

The Batch Commercial Mix is a synthetic batch workload designed to represent multiple types of batch processing often associated with commercial data processing. The different variations allow testing of sequential vs random file access, changing the read to write ratio, generating ″hot spots″ in the data and running with expert cache on or off. It can also represent some jobs that run concurrently with interactive work where the work is submitted to batch because of a requirement for a large amount of disk I/O.

The Commercial Mix operates on 18 physical files of various sizes and record lengths. Each file has 10 logical views using 9 different indexes built over it. The indexes are designed so the physical access to the files can be done sequentially, randomly, or some combination of sequential from a random starting point (skip-sequential).

The commericial mix is executed in a job which is comprised of:

1. **Order Validation simulation.** This is a combination of reading/updating an Order Header file with skip-sequential accesses (blocks of 20 starting at a random location), and then reading through Order Detail, Customer, and Item files in a combination of skip-sequential and random accesses to build a print report of orders in error. The processing associated with this is comprised of a combination of array-handling, moves, simple arithmetic operations, and comparisons using RPG.
2. **Order Processing simulation.** Reads the Header and Detail files in skip-sequential mode and then accesses the Customer and Item files randomly for updates. It then reads through a Shipping file and uses the information to update a series of statistics files and to insert new records into two historical files. It completes by building a printed report of orders processed.
3. **File Copy.** Copies a 100,000 record file into a temporary file.

3.1 Effect of CPU speed on Batch

The capacity available from the CPU affects the run time of batch applications. More capacity can be provided by either a CPU with a higher CPW value, or by having other contending applications on the same system consuming less CPU. The commercial mix was run on two V3R6 models.

Conclusions/Recommendations

- For CPU-intensive batch applications, run time scales inversely with Relative Performance Rating (CPWs). This assumes that the number synchronous disk I/Os are only a small factor.
- For I/O-intensive batch applications, run time may not decrease with a faster CPU. This is because I/O subsystem time would make up the majority of the total run time.
- It is recommended that capacity planning for batch be done with tools that are available for AS/400. For example, BEST/1 for OS/400 (part of the Licensed Program Product, AS/400 Performance Tools, 5716-PT1) can be used for modeling batch growth and throughput. BATCH400 (an IBM internal tool) can be used for estimating batch run-time.

3.2 Effect of DASD type on Batch

For batch applications that are I/O-intensive, the overall batch performance is very dependent on the speed of the I/O subsystem. Depending on the application characteristics, batch performance (run time) will be improved by having DASD that has:

- faster average service times
- read ahead buffers
- write caches

Additional information on DASD devices in a batch environment can be found in Chapter 14, “DASD Performance” on page 237.

3.3 Tuning Parameters for Batch

There are several system parameters that affect batch performance. The magnitude of the effect for each of them depends on the specific application and overall system characteristics. Some general information is provided here.

- **Expert Cache**

  Expert Cache did not have a significant effect on the Commercial Mix batch workload. Expert Cache does not start to provide improvement unless the following are true for a given workload. These include:

  – The application that is running is disk intensive, and disk I/O′s are limiting the throughput.
  – The processor is under-utilized, at less than 60%.
  – The system must have sufficient main storage.

  For Expert Cache to operate effectively, there must be spare CPU, so that when the average disk access time is reduced by caching in main storage, the CPU can process more work. In the Commercial Mix benchmark, the CPU was the limiting factor.

  However, specific batch environments that are DASD I/O intensive, and process data sequentially may realize significant performance gains by taking advantage of the larger memory sizes available on the RISC models, particularly at the high-end. Even though in general applications require more main storage on the RISC models, batch applications that process data sequentially may only require slightly more main storage on RISC. Therefore, with larger memory sizes in conjunction with using Expert Cache, these applications may achieve significant performance gains by decreasing the number of DASD I/O operations.

- **Job Priority**

  Batch jobs can be given a priority value that will affect how much CPU processing time the job will get. For a system with high CPU utilization and a batch job with a low job priority, the batch throughput may be severely limited. Likewise, if the batch job has a high priority, the batch throughput may be high at the expense of interactive job performance.

- **Dynamic Priority Scheduling**

  See 18.2, “Dynamic Priority Scheduling” on page 294 for details.

- **Application Techniques**

  The batch application can also be tuned for optimized performance. Some suggestions include:

  – Consider breaking the application into pieces and having multiple batch threads (jobs) operate concurrently. Being batch jobs are typically serialized by I/O, this will decrease the overall required batch window requirements.

  – Like any application, reduce the number of opens/closes, I/Os, etc. where possible.

  – If you have a considerable amount of main storage available, consider using the Set Object Access (SETOBJACC) command. This command pre-loads the complete database file, database index, or program into the assigned main storage pool if sufficient storage is available . The objective is to improve performance by eliminating disk I/O operations.

  – If communications lines are involved in the batch application, try to limit the number of communications I/Os by doing fewer (and perhaps larger) larger application sends and receives. Consider blocking data in the application. Try to place the application on the same system as the frequently accessed data.

Chapter 4. DB2 for AS/400 Performance

This chapter contains performance information on items that are important to achieving a good overall level of performance for DB2 for AS/400 environments. The information presented here concentrates on performance for applications run locally on DB2 for AS/400, although much of the information can also be used to ensure better levels of performance for applications using remote access to an AS/400. For more detailed information on the performance of remote access to DB2 for AS/400, refer to Chapter 10, "DB2/400 Client/Server and Remote Access Performance" on page 165.

The various parts throughout the first portion of this chapter concentrate on DB2 for AS/400 enhancements made in Version 4 (V4R3, V4R2, and V4R1). These sections each contain a description of the enhancement and highlights of the performance aspects of that item.

The final part of this chapter contains articles that were written in past (Version 3) releases about DB2 for AS/400 performance characteristics.

4.1 Sources of Additional Information

The AS/400 Teraplex Center plays a significant role in verifying the benefits of new technologies for data warehousing operations. In many cases their testing applies to other, general applications of these technologies as well. For a variety of recent test results and additional information on many of the new technologies described in this chapter, see the Teraplex Center′s Internet home page at url

http://www.softmall.ibm.com:80/as400/bi/teraplex/

The AS/400 Systems Performance area maintains an IBM intranet home page (available to IBM personnel and others who have access to the IBM intranet). For a variety of white papers and additional information on many of the technologies described in this chapter, see the Performance section at url

http://ca-web.rchland.ibm.com/perform/perfmenu.htm

For information and guidance about database performance tuning and query performance tuning, refer to the DB2 for AS/400 Database Programming manual (SC41-5701) (appendix on query performance) and the DB2 for AS/400 SQL Programming manual (SC41-5611) (chapter on data management and query optimizer).

4.2 Encoded Vector Indices (EVIs)

In V4R3 a new type of permanent index, the Encoded Vector Index (EVI), can be created through SQL. EVIs cannot be used to order records, but in many cases, they can improve query preformance. An EVI has several advantages over a traditional binary radix tree index.

- The query optimizer can scan EVIs and automatically build dynamic (on-the-fly) bitmaps much more quickly than traditional indexes. For more information on dynamic bitmaps, see the description in their section below.

- EVIs can be built much faster and are much smaller than traditional indexes. Smaller indexes require less DASD space and also less main storage when the query is run.
- EVIs automatically maintain exact statistics about the distribution of key values, whereas traditional indexes only maintain estimated statistics. These EVI statistics are not only more accurate, but also can be accessed more quickly by the query optimizer.

EVIs are used by the AS/400 query optimizer with dynamic bitmaps and are particularly useful for advanced query processing. EVIs will have the biggest impact on the complex query workloads found in business intelligence solutions and adhoc query environments. Such queries often involve selecting a limited number of rows based on the key value being among a set of specific values (eg a set of state names).

When an EVI is created and maintained, a symbol table records each distinct key value and also a corresponding unique binary value (the binary value will be 1, 2, or 4 bytes long, depending on the number of distinct key values) that is used in the main part of the EVI, the vector (array). The subscript of each vector (array) element represents the relative record number of a database table row. The vector has an entry for each row. The entry in each element of the vector contains the unique binary value corresponding to the key value found in the database table row.

The following is an example of how to create an EVI with the SQL CREATE INDEX statement:

```
CREATE ENCODED VECTOR INDEX StateIx
ON CUSTOMERS (CustState)
```

4.3 Parallel Data Loader

The data loader is a new function in V4R3 (also available via PTF on V4R2 and V4R1) that makes loading AS/400 database tables from external data much simpler and faster. The data loader can import fixed-format, delimited, and byte-stream files. A new CL command, Copy From Import File (CPYFRMIMPF), is provided to simplify the process.

After installing and activating the DB2 for AS/400 SMP licensed feature on a multiprocessor AS/400, parallel processing increases the speed of the data loader by approximately ten times over non-parallel methods. With this feature active, DB2 for AS/400 is able to use multi-tasking, rather than just a single task, to load an import file.

The following PTFs are required to provide these functions on earlier releases:

- V4R1M0 PTFs: SF47138 and SF47177 for OS/400
- V4R2M0 PTFs: SF46911 and SF46976 for OS/400

These are available individually and may be available in a cumulative PTF package.

4.4 Parallel Index Maintenance

Parallel index maintenance, supported in V4R3, can be useful to those that have many logical files and indexes defined over a single database file. Every time that a row is inserted, changed or deleted into a database table, all of the indexes and logical files defined over that base table have to be maintained to reflect the latest data change. The parallel index maintenance enhancement allows DB2 for AS/400 to maintain multiple indexes in parallel instead of one at a time as done in previous releases. Note that DB2 for AS/400 will utilize parallel index maintenance only when blocked insert operations are performed on the base database table, there exist at least eight indexes over the table, and the DB2 for AS/400 SMP (Symmetric Multiprocessing) licensed feature is installed and activated. Parallel index maintenance thus allows DB2 for AS/400 to reduce the amount of time it takes to maintain indexes when you are adding lots of new rows to a database table. The data loader and copy file (CPYF) utilities also will benefit from this feature since they utilize blocked inserts.

4.5 Dynamic Bitmaps

Dynamic bitmaps, introduced in V4R2, can improve the performance of certain query operations. A dynamic (on-the-fly) bitmap is a temporary binary structure built against a permanent index. The AS/400 database query optimizer automatically builds dynamic bitmaps when the optimizer determines that dynamic bitmaps will speed up response time. Use of dynamic bitmaps allows the system to perform skip-sequential DASD operations and reduces the need for full-table-scans, thereby reducing database I/O operations and speeding up completion of the affected queries. In addition, DB2 for AS/400′s dynamic bitmap support allows the use of multiple indexes against any particular table in the query (previously limited to at most one index per table). These multiple bitmaps resulting from the use of more than one index are combined into a composite results bitmap using boolean logic.

When the optimizer builds a dynamic bitmap, it sets a bit for every index entry that meets the selection criteria. It can combine (AND, OR) multiple bitmaps into a composite results bitmap. The optimizer uses this final bitmap to retrieve only those records whose bits are set.

Note that single-column indexes can often maximize flexibility for the database administrator. Dynamic bitmaps allow the optimizer to use several of these single-column indexes at once and combine their dynamic bitmaps, using boolean logic, into one bitmap. In this way, ad hoc users of large databases may be able to realize large performance gains without significant impact to the system and with a minimal set of indexes.

4.6 System-Wide SQL Statement Cache

The system-wide SQL statement cache, introduced in V4R2, can improve the performance of programs using dynamic SQL. The system automatically caches dynamic SQL statements. No user action is required to activate or administer this function.

When a dynamic SQL statement that has previously executed is later reexecuted, if the statement is still available in the cache, DB2 for AS/400 can retrieve (rather than construct) key information associated with the cached statement, and thereby reduce the processing resource and the time required to execute the statement again.

4.7 Remote Journal Function

Introduced in V4R2, the remote journal function allows replication of journal entries from a local (source) AS/400 to a remote (target) AS/400 by establishing journals and journal receivers on the target system that are associated with specific journals and journal receivers on the source system. Some of the benefits of using remote journal include:

- Allows customers to replace current programming methods of capturing and transmitting journal entries between systems with more efficient system programming methods. This can result in lower CPU consumption and increased throughput on the source system.
- Can significantly reduce the amount of time and effort required by customers to reconcile their source and target databases after a system failure. If the synchronous delivery mode of remote journal is used (where journal entries are guaranteed to be deposited on the target system prior to control being returned to the user application), then there will be no journal entries lost. If asynchronous delivery mode is used, there may be some journal entries lost, but the number of entries lost will most likely be fewer than if customer programming methods were used due to the reduced system overhead of remote journal.
- Journal receiver save operations can be off-loaded from the source system to the target system, thus further reducing resource and consumption on the source system.

Hot backup, data replication and high availability applications are good examples of applications which can benefit from using remote journal. Customers who use related or similar software solutions from other vendors should contact those vendors for more information.

Test Environment

As mentioned above, remote journal can be run in either synchronous delivery mode or asynchronous delivery mode. In synchronous mode, the source system must wait for a confirmation message from the target system that the journal entries have been received. In asynchronous mode, the source system runs without having to wait for remote journal to finish. In both sync and async modes, remote journal can be in one of two activation modes. If the source journal receivers contain entries that have not been replicated at the time remote journal is started, then remote journal will run in ″catch-up″ mode to transfer these entries to the remote system. Once remote journal catches up with the source system journal entries, it then runs in ″continuous″ mode.

Tests were done in Rochester to evaluate the following aspects of remote journal performance:

- Comparing elapsed times of running catch-up mode using TCP/IP, APPC and Opticonnect for OS/400
- Overall impact of running remote journal in continuous mode in an interactive transaction processing environment using TCP/IP, APPC and Opticonnect for OS/400

For all tests, the source system used was a model 530-2151 with 2.5 GB of memory and 73 GB of DASD (24 arms totalling 65 GB in the system ASP, 4 arms totalling 8 GB in a user ASP). The target system was a model 510-2143 with 1.0 GB of memory and 33 GB of DASD (16 arms totalling 25 GB in the system ASP, 4 arms totalling 8 GB in a separate user ASP). All DASD arms on both systems were unprotected (RAID or mirroring was not used). Both systems were installed with V4R2, and on both systems, the journal receivers were located on the user ASP.

For both the TCP/IP and APPC tests, a 16Mbps token ring was used (model 2629 with 6149 cards on both systems). For Opticonnect, the bus model used was a 2682, with the source system using a 2685 card and the target system using a 2688 card.

Note that the results shown here are for specific environments and configurations running a controlled and repeatable series of tests. The actual results you obtain in your environment may vary from what is presented here, although the conclusions and recommendations made here will be applicable to most customer applications using remote journal.

Remote Journal Catch-Up Mode

For the catch-up mode tests, approximately 3.1 GB of journal entries was transferred to the remote system when remote journal was started. Memory pools were allocated to ensure this resource was not constrained. There was no other activity on either the source or target systems.

Using TCP/IP and APPC, this transfer took just over 26 minutes, while it took about 12 minutes using Opticonnect. These results are as expected given the much faster transfer rates of the Opticonnect bus versus the token ring connection. In all cases, the CPU utilization was low (15% in the Opticonnect case and less than 10% in the other measurements). No other system resources were constrained during these tests.

Remote Journal Continuous Mode

The base run for this test was done using an interactive transaction processing environment with 640 users running about 224,000 transactions per hour on simulated locally attached 5250 workstations. All files that were updated in this environment were journaled to a single local journal receiver on the source system′s user ASP. Commitment control was not used. This environment produces about 3.2 million journal entries per hour with a CPU utilization of about 72% on the source system. Memory and DASD are not constrained in this environment. The response time for this environment is based on one transaction type that accounts for 45% of the total throughput and produces about 35 journal entries per transaction. In this environment, the average response time for this transaction was 0.15 seconds.

For the remote journal runs, all journal entries produced on the source system were replicated on the target system. Both asynchronous mode and synchronous mode were tested on each of the three different communications protocols. In the remote journal runs, the target journal receiver was located on the user ASP on the target system, and again memory and DASD were not constrained. There was no other activity on the target system.

When remote journal with asynchronous mode was added to the base environment, the results show that for all three protocols, the impact to response time was minimal (about 0.05 seconds) with a corresponding drop in throughput of less than 1%. The increase in CPU utilization on the source system ranged from 2-7%, and the token ring and bus utilizations were very reasonable (less than 15% in all cases). The target system in all three environments showed a CPU utilization of 2-3% with low levels of disk arm utilization.

When remote journal with synchronous mode was added to the base environment, the Opticonnect run showed the least impact to response time (increase of about 0.15 seconds) with a corresponding drop in throughput of about 1.5%, while TCP/IP and and APPC showed an increase of about 0.3 seconds with a drop in throughput of about 3%. In all three environments, the CPU overhead per transaction from adding remote journal was 7-8% on the source system. The token ring utilization in the APPC and TCP runs was about 15-20%, and the bus utilization was also low in the Opticonnect run. The CPU overhead on the target system was about 5-7%, and DASD utilizations on the target system user ASP were again low as they were in the async mode runs.

Conclusions and Recommendations

- For most interactive environments, adding remote journal should not result in significant degradations to response time or resource utilization on either the source or target system. Although you can expect a more noticeable increase in response times when using synchronous mode, the tests above show that the amount of the increase should still be reasonable. In sync mode, the amount of increase will be relative to the number of journal entries produced by an average transaction.

  One other item to consider is that the interactive transactions described here spend over 70% of their time doing database activity, with 20% or so spent in application code. In many customer environments, the average transaction spends a much higher percentage of time in application code and other areas and a much lower percentage in database activity. Overall, this means that for many customer applications, the overall impact from adding remote journal may be less than it was in the tests described here.

- There are several factors to consider when deciding what sort of remote journal setup you will choose:

  1. Whether or not you can afford loss of some journal entries in the event of a system failure. Synchronous mode will guarantee no journal entries are lost, while asynchronous mode may lose some entries (although the

number will be relatively low compared to the total number of journal entries being replicated).

2. In general, Opticonnect offers noticeable advantages over APPC or TCP/IP alternatives via token ring, including less of an impact to response times (particularly when using sync mode) over a broader range of throughput and also higher overall capacity. For customers who already have Opticonnect installed and have additional capacity available, this would be a logical choice for implementing remote journal. However, users who do not currently have Opticonnect need to balance the added cost of this product versus what can be accomplished over a token ring using APPC or TCP/IP. For example, if the token ring has enough capacity for your level of remote journal and you either use async mode or can afford the response time increase with sync mode, then using APPC or TCP/IP with the token ring would also be a logical choice.

3. Another item to consider for remote journal is DASD performance, particularly when using Opticonnect. Since Opticonnect allows a much higher transfer rate and capacity without bottlenecking the bus, the performance of the DASD arms where the journal receiver is located can become a problem in very heavy journaling environments, particularly if the arms are RAID protected. In heavy remote journal environments, Opticonnect can achieve transfer rates up to 1 GB per second when the journal receivers on both systems are located on unprotected DASD arms (no RAID or mirroring). If RAID protection is used in the same environment, the DASD arms become the bottleneck. If protection is required, using mirrored arms for the journal receiver DASD will provide significantly better performance than RAID protection. However, unprotected DASD arms will allow the highest overall transfer rates and best overall performance when using Opticonnect for remote journal. Whatever implementation is used, DASD arm performance should still be monitored using the performance monitor and/or other tools such as the WRKDSKSTS command.

   In most cases involving high transfer rates via TCP/IP or APPC, the communications line or IOP will tend to become the bottleneck prior to the journal receiver DASD arms becoming overutilized. However, it is still a good idea to monitor the performance of these arms to ensure the best overall results.

4. In catch-up mode, performance will be significantly better using Opticonnect than with TCP or APPC due to the higher transfer rates and overall higher capacity of the Opticonnect bus. This should also be considered when determining what medium to use in your overall remote journal implementation.

5. The measurements done here show the impact of adding remote journal to an application that is doing local journaling without commitment control. Similar results can be expected when adding remote journal to customer applications that use commitment control.

- If remote journal will be used during execution of a batch job, it is recommended that asynchronous mode be used. The reason for this is that in the event of a system failure, most customers choose to start the batch job over from the beginning instead of restarting the batch job partway through after recovering to that point using the remote journal. Because of this, the additional overhead of using synchronous mode will result in longer elapsed times with no additional benefits.

- As noted several times in this section, it will be important to monitor your resource utilizations on the source and target systems both prior to and after implementing remote journal. Key resources to consider include CPU utilization, line utilization, IOP utilizations, and DASD arm performance, as well as overall response times. These items can be monitored effectively using data collected from the performance monitor tool.

Additional Sources of Information

Customers interested in using the remote journal function should read the chapter on remote journal in the V4R2 version of the *OS/400 Backup and Recovery Guide*, SC41-5304. This chapter contains additional information on functional and performance aspects of remote journal.

4.8 Parallel Index Build

Parallel index build, introduced in V4R1, has several important uses. This function can be used when the DB2 for AS/400 SMP feature is installed and made active. Following is a list of its uses.

- Speeding up the process of building a temporary index, when the query optimizer cannot locate an appropriate index and determines to build a temporary index to execute a particular query. This function is most beneficial in data warehousing environments housing large database files.
- Speeding up the process of building permanent indexes, particularly when built over large database files. Tests at the AS/400 Teraplex Center show that parallel index build, after the loading is complete, is the fastest way to load data and build an index, and is preferred to building indexes while data is being loaded.

V4R3 note: In V4R3, the new encoded vector indexes, as well as the traditional binary radix tree indexes, are supported by the parallel index build function.

4.9 Version 3 DB2 for AS/400 Performance Information

Although much of the information in this section is still applicable to DB2 for AS/400, there may be portions of the articles that are no longer accurate due to changes made since the article was written. As of V4R3, the following general comments should be considered when using the information in this section.

- Prior to V4R1, *MAX4GB was the default value when creating any new index (as discussed below in “Enhanced Index Support for DB2/400” on page 49). Starting in V4R1, the default has been changed to *MAX1TB. In addition, the potential performance concerns with *MAX1TB indexes have been alleviated such that *MAX1TB is now the recommended size for all indexes (hence the change to the default value). The only concern with using *MAX1TB is that indexes created with this value can be up to 15% larger in size than if

*MAX4GB is used, so it may be best to monitor available DASD space when converting or creating these types of indexes.

- In addition to the SQL improvements discussed in sections “DB2/400 SQL and Query Information Improvements” on page 51 and “DB2/400 SQL” on page 52, there have been additional changes made in subsequent releases that in many cases will allow well-tuned SQL applications on the AS/400 to be competitive with these same applications run on other hardware platforms. Although there are too many changes to list or discuss here, it is recommended that customers currently running their SQL applications on V4R1 or prior releases consider moving to V4R2 or a later release to take advantage of these performance improvements. Users should be able to take advantage of these improvements with no changes to or recompiles of existing SQL applications.
- In the “DB2/400 SQL” on page 52 section, several of the additional sources of information listed may no longer be available under the same document numbers or titles. The titles listed in this section were accurate as of V3R6 and V3R7, but may have since been moved or deleted. It may be best to contact your IBM representative if you need to know what current sources of information are available concerning SQL performance.

DB2/400 Enhancements in V3R6

Enhanced Index Support for DB2/400

The index support was enhanced in V3R6 to support larger indexes and to reduce index seize contention. Now rather than a size limitation of 4 Gigabytes (GB), each index can be as large as One Terabyte (TB). This limit is related to the size of the index and not the number of entries.

To take advantage of the increased index size capabilities, a new parameter, ACCESS PATH SIZE, has been added to the Create Physical File (CRTPF), Create Logical File (CRTLF), and Change Physical File (CHGPF) commands. If you want to allow growth beyond 4 GB on a keyed physical file that already exists, you can use the Change Physical File (CHGPF) command specifying ″*MAX1TB″ for the ACCESS PATH SIZE parameter. If you want to change a logical file to the larger limit, you would delete the existing logical file and create a new logical file specifying ″*MAX1TB″ for the ACCESS PATH SIZE parameter. When creating new files, physical or logical, the default is *MAX4GB. You must specify *MAX1TB if it is required. It is recommended, though not required, that all access paths for a file be of the same type.

Note: The procedure described above will not work for files that have UNIQUE *YES specified. For these files you will have to create a second file with the ACCESS PATH SIZE specified at *MAX1TB and copy the first file to the second. The first file can then be deleted and the second file renamed.

In addition to the capability to create larger files, the algorithm used for seizing indexes has been enhanced. Indexes are now seized at the index page level. Therefore, for those workloads where there is a high-level of concurrency on a particular file or access path, the new algorithm will significantly reduce the contention resulting in significant performance improvements. This is particularly applicable to multi-processor systems and indexes with a high number of records being added. In order to take advantage of this change, the index must be created/changed as specified above.

Most customers will be best served by specifying the default of *MAX4GB for the ACCESS PATH SIZE parameter. This will in general provide better performance. Also if files will be moved to a prior release, the index may need to be rebuilt or the save of the file will not work if the index was created with *MAX1TB.

It is recommended that *MAX1TB be specified only where it is needed to allow for the larger file size or where there is high contention on access paths. If an index is changed to *MAX1TB and there is not high contention on the index, it may result in additional overhead. One measurement on a uni-processor system where this was the case resulted in a slight (less than 5%) slowdown. The amount of overhead would depend on, but not limited to, number of files, size of files, and access patterns.

How to determine when to switch for performance

To determine if the contention on your system is at a level that changing to a *MAX1TB access path will improve performance you will need to collect data using the Performance Monitor. This can be collected using the command STRPFRMON (Start Performance Monitor). When issuing this command, the interval value should be changed from 15 minutes to 5 minutes and the trace data collected should be changed from *NONE to *ALL. Data should be collected for at least 30 minutes and should be collected during ″peak″ activity. Please note that when the Performance Monitor is ended the trace data will automatically be dumped to DASD and performance could be degraded while this data is being transferred. This can be avoided by sepcifying Dump Trace *NO when starting the monitor and at a later time, when the system is not as busy, issuing the DMPTRCDTA command.

If the number of Seize Conflicts Per Second on the Component Report (created by issuing the PRTCPTRPT command or using the menu options from the PERFORM menu) is greater than 140 then you can in many cases benefit from changing to files with the ACCESS PATH SIZE parameter set to *MAX1TB. The method to determine which files should be changed is discussed in the next paragraph.

Once the trace data has been saved (either when the monitor ends if Dump Trace *YES was specified or after issuing the DMPTRCDTA command) a transaction report should be printed by either issuing the PRTTNSRPT command or using the menu options from the PERFORM menu. The SUMMARY OF SEIZE/LOCK CONFLICTS BY OBJECTS section of this report shows the number of seize conflicts and the number of lock conflicts by object. This list will show conflicts on both data spaces (ie. file data) and data space indexes (ie. access paths). If the majority of seize conflicts are on one or two data space indexes, those are the candidates for the larger access path size parameters. The files most likely to fall in this range are those that are accessed by different users at the same time, some for inserting into the file, some for updating records in the file and some for reading the records in the file.

Note: For more detail on how to perform the functions above and what the reports contain as well as how to interpret the data, please consult the PERFORMANCE TOOLS/400 GUIDE publication, SC41-4340-00.

DB2/400 SQL and Query Information Improvements

In V3R6, there have been several enhancements made to improve performance for SQL queries. In addition, more information is now available to help users analyze performance for all queries.

1. In V3R6, there is now more information provided to query users to help them analyze and improve the performance of their queries:

    - Index advisor messages have now been added to the messages that are generated in the joblog when running a query in a job in DEBUG mode. These messages will indicate how an index could be constructed that would be optimal for the performance of that query. Note that the information provided is generally most useful for queries that involve a single file or for the primary file in a join query.

    - There is a new database performance monitor function available via the STRDBMON (Start Database Monitor) command. This monitor will provide detailed information on all DB2/400 queries, such as CPU, I/O, elapsed time, description of the query, etc. The information is placed in a database file where it can be readily queried. The data provided by this function can provide valuable information for performance analysis of any DB2/400 query. For more information on the database monitor, refer to the *DB2 for OS/400 Database Programming* guide.

2. Support has been added to allow more types of join operations such as outer joins and exception joins. Although users could previously construct queries involving UNIONs to do outer joins, this was often cumbersome to do. The new SQL join syntax now gives users the ability to easily code these types of queries, often with significantly improved performance versus the previous alternative methods. For more information on the new syntax, refer to the *DB2 for OS/400 SQL Programming* guide, (SC41-4611).

3. The new join syntax for SQL now also gives users the ability to specify the order in which they want files to be joined. This can help improve performance for join queries where the optimizer is not choosing the optimal order in which to do the join. More information on this is available in the above mentioned SQL Programming guide.

4. Prior to V3R6, all SQL cursors within a given program/module operated under the commitment control level specified for that program when it was compiled. Now, however, a new WITH clause has been added to the SQL SELECT statement to allow specific cursors to run under the desired level of commitment control. For example, if an SQL program is running with a commit level of *ALL but there are read-only cursors in the program that do not need any commitment control, you can add the WITH NC clause to the SELECT statements to have these cursors run under a level of *NONE. Other levels that can be specified are UR (for *CHG), CS (for *CS) and RS (for *ALL). In many cases, a significant improvement in performance can be realized when this type of change is made to cursors that previously had been running under a more stringent level of commitment control than was necessary.

5. Support has been added to the ALTER TABLE SQL command that gives users the ability to easily add new fields and delete/change existing fields in any database file. Although users could do this in the past by deleting the old file and recreating it with a new format, this also meant that any views

and indexes over the file had to be rebuilt as well, which could take a long time to complete. With the new support, the existing database file is copied to another file with the new format, and the indexes over the file are not rebuilt as long as no key fields are being altered. Although it still may take a while to do this copy, altering a database file′s format with ALTER TABLE should be considerably faster than what the user had to do previously.

6. Prior to V3R6, a UNION ALL operation that did not specify an ORDER BY always generated a temporary file containing the results from each of the SELECT statements. This also meant that the ODP for this UNION was not reusable, which resulted in a full open and close each time the UNION was run. In V3R6, this type of operation now operates with ′live′ data, i.e., the results from the second SELECT are not read until all the rows from the first SELECT have been read. This change will in most cases result in significantly improved response times for the first rows returned from the UNION since these rows can be returned immediately without having to wait for the entire temporary file to be filled with results from all the SELECTs. Also, SQL is now able to make the ODP for this operation reusable, which also improves response time significantly.

7. In previous releases, the ODP for a cursor that contained a LIKE clause with a host variable mask was not reusable, which meant a full open was required each time the query was run. In V3R6, the ODP will be reusable if the value in the host variable mask is of the form ′XXXX%′ and the NUMBER of constant characters in the mask stays exactly the same between each run of the query. In the case shown here, the contents of the XXXX constant part may be changed, but the number of constant characters (4) must remain the same and there cannot be anything else in the mask. If these rules are adhered to, users can significantly improve the performance of this type of SQL query.

8. Queries such as UNIONs, subqueries and joins may in some cases specify the same view in each SELECT specified in the query. Prior to V3R6, running these types of queries resulted in the view being evaluated multiple times, once for each time it was specified. In V3R6, the view in cases like this is evaluated only once and the results are used by each SELECT in the query. This change can result in a noticeable improvement in performance for this type of query, particularly if each evaluation of the view is costly.

DB2/400 SQL

DB2/400 Structured Query Language (SQL) support provides the user with an additional means of accessing data within an OS/400 relational database. This support provides several advantages in terms of flexibility, productivity and portability between various database platforms. However, prior to using SQL, users should also consider what level of performance they can expect when using this product. This section will provide general information on the performance of SQL to help users better determine what this level of performance will be.

This section is not intended to be a complete guide to SQL performance. For many users, other items of interest will include the SQL optimizer, performance tips and techniques, database design, etc. It is important that users take advantage of SQL performance tips and techniques as much as possible when writing an application using SQL. In particular, it is very important to properly construct and use indexes to provide the best overall performance for SQL. This, along with many other tips and techniques, can be found in the sources listed in “Additional Sources of Information” on page 54.

In V3R6, enhancements have been made to SQL that will help many users obtain an overall performance improvement. For a description of these enhancements, refer to “DB2/400 SQL and Query Information Improvements” on page 51.

Performance of SQL versus Native DB

When current users of AS/400 native language I/O (i.e., COBOL/400, RPG/400, etc.) are considering using SQL in their applications, one key item that needs to be considered is what level of difference in performance to expect when making this change.

Note that this section will not provide detailed information on the difference in performance between native and SQL for specific types of I/O operations. However, this type of information can be found in Chapter 5 of the document entitled *″SQL/400 - A Guide for Implementation″* (GG24-3321-01). Also, Chapter 3 of the second version of this document (GG24-3321-02) contains a section entitled ″When to Use SQL″ that provides additional information and guidelines on this subject.

It is difficult to predict how SQL will compare to native DB access for a given application. Generally, SQL will use anywhere from 10-30% more CPU than native, although this may vary considerably depending on the type of operations being done. For example, SQL shows considerably more overhead than native when operating on one record at a time, such as an OPEN-FETCH-UPDATE WHERE CURRENT OF-CLOSE sequence. However, for more complex operations or for operations involving a lower number of SQL statements, SQL will in many cases show relatively equal or better performance levels than native.

Note that the difference between SQL and native performance is usually in terms of CPU (the amount of I/O that occurs is generally about the same for native and SQL for similar functions). Note, however, that on systems where the CPU utilization has not reached the knee of the performance curve, a difference in CPU of 10-30% per transaction will not result in a large difference in response time. Beyond the knee, however, the response time difference may grow considerably.

Other Performance Notes

- Generally, the use of SQL will result in significantly increased memory requirements when compared to similar native DB operations. This is mainly due to the additional internal structures and program automatic storage required by SQL to maintain optimum performance levels, as well as the fact that SQL cannot share ODPs across or within applications as native DB can. However, it is important to remember that the extra memory required can vary widely from application to application and is mostly dependent on the complexity of the application. Simple applications involving only a small number of I/O operations may require little additional memory for SQL, but for complex applications involving many I/O operations the difference in memory requirements between native and SQL can be significant. When using SQL, users should monitor memory utilization using the AS/400 Performance Tools to better determine if additional memory will be required. In addition, support is available through the Quicksizer tool on HONE to help users size their system for SQL.

- Some SQL users may notice an increase in the amount of auxiliary storage used once their applications begin running. The main reason for this is again largely due to the number and complexity of the ODPs and other internal structures that are maintained by SQL for each individual user for optimum performance. It is important to remember that this additional storage requirement is only temporary, i.e., when the user′s job ends, the storage will return to normal levels. However, for systems where auxiliary storage usage is already high, some evaluation of the number of active SQL users and their storage requirements may be needed prior to any large scale implementation of a given SQL application.
- If memory and/or auxiliary storage is a concern, there are ways to reduce consumption of these resources for SQL applications. Following are some methods that can be used:
  - In a given SQL program, combine any duplicate or like SQL statements into one statement in an internal procedure, and then call that procedure as needed. This will help reduce the number of ODPs for that program.
  - If an internal procedure containing SQL statements is duplicated in several different SQL programs, the number of ODPs can be reduced by placing this internal procedure into a separate SQL program and then calling that program as needed. However, the user needs to be careful when doing this to ensure that the ODPs in this common program are reusable across different invocations of the program. Also, since external calls will cause some performance degradation, this also needs to be considered prior to implementing this type of change.
  - Some user applications ″pre-open″ all their SQL ODPs in order to avoid full opens when the SQL statements are issued. Although this will provide good performance in many cases, there may be ODPs that are pre-opened but rarely used. If this is true, the user may want to be more selective about which ODPs are pre-opened and which are left to be opened when the SQL statement is first issued.

Additional Sources of Information

There are several other sources of information available that the user should obtain to gain a better understanding of SQL and OS/400 database operations, as well as understanding how to properly code SQL functions in order to optimize performance. Following is a list of these sources.

- *SQL/400 - A Guide for Implementation*, GG24-3321

  There are three versions of this document currently available (01, 02 and 03). All versions contain key information and concepts users attempting to gain a better understanding of SQL operations.

  This information is also available on HONE. Please refer to HONE item number RTA000011842.

- *Quicksizer* support for SQL

  Available on HONE to help users size their system for using SQL.

- *System Selection Guide*

  Available on HONE under the title ″System Selection Guide″. Hardcopy editions of the U.S. version and the worldwide version are also available.

- *DB2 for OS/400 SQL Programming*, SC41-4611
- *DB2 for OS/400 Database Programming*
- *SQL/400 Programmer′s Guide*, SC41-9609
- *SQL/400 Reference*, SC41-9608
- *AS/400 Database Guide*, SC41-9659
- *New Products Planning Information (NPPI)*, GA41-0007
- FTN broadcasts entitled ″SQL Performance Technical Update″

  There are several versions of this broadcast currently available on videotape. Each version covers SQL improvements made in each of the past several releases.

- Classes and workshops available to help users understand SQL programming and OS/400 database design and coding. One such recommended workshop is the ″SQL Performance Workshop″.
- SQL presentations from U.S. and European COMMON conferences.

Query Management

OS/400 Query Management (QM) is the AS/400 implementation of the SAA Common Programming Interface (CPI) Query. It provides a common method for accessing data and reporting the results from a relational database across the different platforms allowed by SAA. QM is also a very powerful and flexible reporting tool that provides users with the ability to design and format printed reports that result from the processing of a query. Queries can be included in programs written in RPG, COBOL, or C language and also can be run from within CL programs, giving programmers flexibility in how they set up the environment.

As a general rule, QM queries perform noticeably slower than similar functions that are done via AS/400 Query or embedded SQL because of the additional CPU used by QM (disk I/O characteristics are similar to those of AS/400 Query and SQL). For example, generating large reports via QM in most cases is significantly slower than AS/400 Query, and transaction processing with QM compares poorly with other alternatives such as static SQL. The exception to this rule is in processing summary-only functions such as AVG, COUNT, MIN, MAX and SUM. For this type of function, QM offers equal or better performance than AS/400 Query and performance levels similar to those when using SQL. Also, QM is optimized toward producing and displaying the first screen of output, so response times for this type of activity may be better than that for generating reports from QM.

QM should generally not be considered as an end-user tool. However, the SQL/400 Query Manager is available as an end-user type of interface for QM. Since it sits on top of QM and uses QM support, performance for this product will be similar to that of QM.

In general, customers who are considering the use of OS/400 Query Management need to weigh the functional advantages and flexibility (particularly in the area of portability across various SAA platforms) against the overall performance level of this product. Doing this will help decide if QM is a viable alternative to other AS/400 query products.

Referential Integrity

In a database user environment, there are frequent cases where the data in one file is dependent upon the data in another file. Without support from the database management system, each application program that updates, deletes or adds new records to the files must contain code that enforces the data dependency rules between the files. Referential Integrity (RI) is the mechanism supported by DB2/400 that offers its users the ability to enforce these rules without specifically coding them in their application(s). The data dependency rules are implemented as referential constraints via either CL commands or SQL statements that are available for adding, removing and changing these constraints.

For those customers that have implemented application checking to maintain integrity of data among files, there may be a noticeable performance gain when they change the application to use the referential integrity support. The amount of improvement depends on the extent of checking in the existing application. Also, the performance gain when using RI may be greater if the application currently uses SQL statements instead of HLL native database support to enforce data dependency rules.

When implementing RI constraints, customers need to consider which data dependencies are the most commonly enforced in their applications. The customer may then want to consider changing one or more of these dependencies to determine the level of performance improvement prior to a full scale implementation of all data dependencies via RI constraints.

Triggers

Trigger support for DB2/400 allows a user to define triggers (user written programs) to be called when records in a file are changed. Triggers can be used to enforce consistent implementation of business rules for database files without having to add the rule checking in all applications that are accessing the files. By doing this, when the business rules change, the user only has to change the trigger program.

There are three different types of events in the context of trigger programs: insert, update and delete. Separate triggers can be defined for each type of event. Triggers can also be defined to be called before or after the event occurs.

Generally, the impact to performance from applying triggers on the same system for files opened without commitment control is relatively low. However, when the file(s) are under commitment control, applying triggers can result in a significant impact to performance.

Triggers are particularly useful in a client server environment. By defining triggers on selected files on the server, the client application can cause synchronized, systematic update actions to related files on the server with a single request. Doing this can significantly reduce communications traffic and thus provide noticeably better performance both in terms of response time and CPU. This is true whether or not the file is under commitment control.

The following are performance tips to consider when using triggers support:

- Triggers are activated by an external call. The user needs to weigh the benefit of the trigger against the cost of the external call.
- If a trigger is going to be used, leave as much validation to the trigger program as possible.
- Avoid opening files in a trigger program under commitment control if the trigger program does not cause changes to commitable resources.
- Since trigger programs are called repeatedly, minimize the cost of program initialization and unneeded repeated actions. For example, the trigger program should not have to open and close a file every time it is called. If possible, design the trigger program so that the files are opened during the first call and stay open throughout. To accomplish this, avoid SETON LR in RPG, STOP RUN in COBOL and exit() in C.
- If the trigger program opens a file multiple times (perhaps in a program which it calls), make use of shared opens whenever possible.
- If the trigger program is written for the Integrated Language Environment (ILE), make sure it uses the caller′s activation group. Having to start a new activation group every time the time the trigger program is called is very costly.
- If the trigger program uses SQL statements, it should be optimized such that SQL makes use of reusable ODPs.

In conclusion, the use of triggers can help enforce business rules for user applications and can possibly help improve overall system performance, particularly in the case of applying changes to remote systems. However, some care needs to be used in designing triggers for good performance, particularly in the cases where commitment control is involved.

System-Managed Access-Path Protection (SMAPP)

Description

System-Managed Access-Path Protection (SMAPP) offers system monitoring of potential access path rebuild time and automatically starts and stops journaling of system selected access paths dynamically in order to meet a specified access path recovery time.

The default system wide access path recovery time for SMAPP is 150 minutes. This means that SMAPP protects the system so that there will be no more than 150 minutes of access path rebuild time during an IPL after an abnormal termination. Users can easily alter this value through the EDTRCYAP (Edit

Recovery for Access Paths) or CHGRCYAP (Change Recovery for Access Paths) commands. SMAPP takes over the responsibility of providing the necessary amount of protection. No user intervention is required as SMAPP will manage the entire journal environment.

For systems with user auxiliary storage pools (ASPs), the recovery time can be specified for each ASP rather than one number for the entire system. This granularity allows the users to specify recovery time according to the criticality of the data on these ASPs. However, it is not recommended to specify target access path recovery times for both the entire system and individual ASPs.

For more information on SMAPP, see the *Backup and Recovery - Advanced Book* (SC41-3305).

SMAPP Impacts on Overall System Performance

The overhead of SMAPP varies from system to system and application to application due to the number of variables involved. For most customers, the default value of 150 minutes will minimize the performance impact while at the same time providing a reasonable and predictable recovery time and protection for key access paths. For many environments, even 60 minutes of IPL recovery time will have negligible overhead. Although SMAPP may start journaling access paths, the underlying SMAPP support is designed to be much cheaper in terms of performance than explicit journaling support.

Note that as the target access path recovery time is lowered, the performance impact from SMAPP will increase. You should balance your recovery time requirements against the system resources required by SMAPP.

Although the default level of SMAPP protection will be sufficient for most customers, some customers will need a different level of protection. The important variables are the number of key changes and the number of unprotected access paths. For those users who have experienced abnormal IPL access path recovery longer than 150 minutes it is advisable to experiment by varying the amount of protection. Too much protection causes undue CPU consumption whereas too little protection causes undesirable IPL delay. Customers may need to decide on an optimum SMAPP setting by understanding their system requirements and experimenting to find what value meets these requirements.

There is some help for those who want to experiment. The component report produced by the licensed program *Performance Tools/400* has a database journaling summary. It has information that can help explain the effects of various SMAPP settings. This information is also available to all customers without this licensed program except it takes a little work to query the information (see the chapter titled *Collecting Performance Data* in the *Work Management Guide*).

Users may also experience more DASD usage if they are explicitly journaling their physical files and SMAPP starts journaling for the access paths to the same user journal. However, this increase may be lessened by using the RCVSIZOPT(*RMVINTENT) option on the CRTJRN or CHGJRN command. This will cause the system to remove internal entries used only for IPL recovery when they are no longer needed.

There will be some customer environments (such as those having a tight batch window) where no additional performance overhead can be tolerated. For these environments, it recommended that the SMAPP setting be changed to a much higher number or *NONE prior to the batch window and then changed back to the default/chosen value during transaction-heavy hours.

If ANY overhead at all cannot be tolerated, SMAPP can be turned off completely (special value *OFF). In this mode, there is no performance overhead, but there is also no idea of how exposed the system is. Also, to turn SMAPP back on, the system must be in a restricted state. Therefore, it is not advisable to turn SMAPP *OFF. The differences between SMAPP *NONE and SMAPP *OFF are:

- SMAPP *NONE allows SMAPP to monitor the system exposure without journaling access paths.
- You do not have to be in a restricted state to change from SMAPP *NONE to any other setting.

Miscellaneous Notes

1. SMAPP has no performance impact when you run applications with no access paths or those that do not make any key changes.
2. If SMAPP has a noticeable impact to performance, it will generally be in terms of increased CPU utilization and/or increased asynchronous IO activity. In most cases, SMAPP will have little effect on the the amount of synchronous IO.
3. The system starts to journal ALL access paths when SMAPP is set at *MIN (minimum access rebuild time during IPL or maximum protection). In some environments, the overhead of *MIN can result in a significant impact to overall system performance. For this reason, *MIN is not a recommended setting. If you have several small access paths that have many key changes, you are better off paying the small price of rebuilding them in the IPL following an abnormal termination (which is not frequent) than paying the runtime overhead of maximum SMAPP protection.
4. SMAPP and explicit journaling (of physical files and/or access paths) can coexist and are compatible with each other.
5. If SMAPP decides to journal an access path for a physical file that is currently not being explicitly journaled, SMAPP must journal both the physical file and the access path. The impact from this change can be noticeable to an application′s performance. However, if SMAPP also decides to journal more access paths for the physical file, the added cost of journaling each additional access path will be less than the impact from journaling the first access path.

Journaling and Commitment Control

This section provides performance information and recommendations for DB2/400 journaling and commitment control.

Journaling

The primary purpose of journal management is to provide a method to recover database files. Additional uses related to performance include the use of journaling to decrease the time required to back up database files and the use of access path journaling for a potentially large reduction in the length of abnormal IPLs. For more information on the uses and management of journals, refer to the *AS/400 Backup and Recovery Guide*

- The addition of journaling to an application will impact performance in terms of both CPU and I/O as the application changes to the journaled file(s) are entered into the journal. Also, the job that is making the changes to the file must wait for the journal I/O to be written to disk, so response time will in many cases be affected as well.

  Journaling impacts the performance of each job differently, depending largely on the amount of database writes being done. Applications doing a large number of writes to a journaled file will most likely show a significant degradation both in CPU and response time while an application doing only a limited number of writes to the file may show only a small impact.

- The impact to performance from adding journaling can be reduced by locating the journal receiver on a separate user ASP. Doing this will generally reduce the seek time required to access the disk arms for journal I/O which will in turn help reduce the impact to end user response time. It will also lessen the impact to the disk arms located on the system ASP.

  When using a separate user ASP for journal receivers, it is important to consider the number of disk actuators available in the ASP. Customer environments with heavily used journal receivers located in a user ASP that consists of a single disk actuator may actually reach a limit to performance because of the high usage of this single actuator. In this case, it would be better to have multiple disk actuators available in the user ASP so that DB2/400 journaling support can interleave journal entries over the multiple actuators, thus reducing contention for any one single disk arm. Doing this may result in an improvement in response time and in overall system throughput. However, it is important to note that although adding an actuator may provide a significant improvement in performance, each additional actuator added beyond this will improve performance to a lesser degree. Once the utilization of the actuators is low, adding more actuators will not improve performance.

  Having two or more journal receivers located on the same user ASP and having them in use at the same time may not take full advantage of the performance gains seen by isolating a single journal receiver on the User ASP since the seek distance on the actuator increases as the journal entries are written to the two receivers.

- Tracked asynchronous I/O is used to write the journal entries to disk. The use of this type of I/O allows the journal support to determine on a process by process basis, which processes need to wait for the I/O to complete and which are allowed to continue. However, by using tracked asynchronous I/O, all I/O operations to a journal receiver now appear in performance reports as asynchronous even though the process may actually be waiting for the I/O operation to complete. This could cause the Capacity Planning tools to recommend a smaller configuration than is necessary. This should be considered if a measured profile is created for purposes of future system capacity planning.

Commitment Control

Commitment control is an extension to the journal function that allows users to ensure that all changes to a transaction are either all complete or, if not complete, can be easily backed out. The use of commitment control adds two more journal entries, one at the beginning of the committed transaction and one at the end, resulting in additional CPU and I/O overhead. In addition, the time that record level locks are held increases with the use of commitment control. Because of this additional overhead and possible additional record lock contention, adding commitment control will in many cases result in a noticeable degradation in performance for an application that is currently doing journaling.

- There are instances where adding commitment control can result in improved response times for an application doing journaling. As stated before, journaling alone means that the journal entries for changes to the file are written synchronously to disk. However, under commitment control, most journal entries are written to disk asynchronously. Only the final journal entry of the commit cycle (along with any entries of the cycle that have not yet been written to disk) are written synchronously. Because of this, applications may no longer have to wait for each journaled change to be written, which can result in reduced response times. The amount of improvement will depend mainly on the number of journal entries within the commit cycles - the more entries per cycle, the greater the potential for improving response time over journaling alone. For example, adding commitment control to a dedicated batch job that is currently doing journaling could potentially improve the job run time if there are a large number of changes to the physical files being journaled.

  It is important to remember that the potential for improving response time by adding commitment control is also largely affected by overall system resource utilization. Environments that are showing high CPU or disk utilization or have constrained memory will in most cases show a degradation in performance from adding commitment control because of the additional CPU and I/O required. Also, adding commitment control can result in record level lock contention between jobs, which can also affect response time. Given the number of variables involved, a test run is highly recommended prior to adding commitment control for the purpose of improving performance in a production environment.

Date/Time Fields

The support for date and time fields in DB2/400 provides a number of advantages for the end user:

- Programmer productivity may be improved when an application requires calculations on date or time fields. New functions can be added more easily.
- Since the date and time data is stored in an internal format and converted on retrieval, the same underlying data can be viewed in different formats based on the needs of the application.

- Because the internal format reflects the sequential nature of time, it can be easily used to sort data in terms of sequence. For example, if a file currently contains a date in MMDDYY format, special application processing is required to sort it in YYMMDD sequence. This application processing is not needed when the date is stored in internal format.
- Some applications may achieve small savings in file size and DASD requirements since the internal formats are generally smaller than external formats.

The use of DB2/400 date/time support in many cases will result in additional CPU resource being used. Generally, the increase will be less than 10% but is dependent upon the number of calculations and the number of date and time fields being accessed. Time fields will usually show minimal impact while date and timestamp data types may show more of an effect on performance.

Note that in terms of performance, DB2/400 date/time support is generally better than or equal to other generalized routines that support many different date/time formats. However, when compared to date/time routines that handle only very specific date/time formats, DB2/400 date/time support may have higher CPU requirements.

When using date/time support in products such as AS/400 Query and Query Management, the amount of additional CPU required will vary. In many instances, the impact will be minimal and may even show a small reduction in CPU versus previous methods of providing this type of support. For example, report breaks on date fields under AS/400 Query will in many cases provide comparable performance to using packed data for dates. However, there are certain cases where the use of date/time support can result in significant performance overhead:

- When replacing the use of zoned decimal data for dates
- When adding a result field calculations to a query (such as adding 90 days to a date)
- Report breaks on date fields under Query Management (compared to the use of packed data for dates)

Overall, DB2/400 date/time support can provide many functional advantages to user applications without a significant impact to performance. However, the user should exercise some caution when implementing this support in order to minimize this impact.

Null Values

DB2/400 provides support for the use of null values in any field in any file. For a more detailed description of null value support, refer to the SQL/400 Reference or the SQL/400 Programmers Guide.

The performance impact from using null value support will vary depending on the number of fields declared as null capable and on the number of records being accessed. For example, when a user even changes only one field in a file to be null capable, there will be a slight increase in the CPU resource required to either insert records into or read records from this file. The amount of the increase should be about the same whether or not the null capable field actually contains null values. Also, as the number of null capable fields in a given file record format increases, the CPU required to process each record will also increase. For operations such as AS/400 Query, Query Management and SQL/400 queries that select all the fields from a large number of records, the impact of adding null capable fields to the file can be significant in terms of increased CPU.

Because of the potential impact, users need to be somewhat careful in what files null capable fields will be used and in deciding how many fields will be null capable. Although null capable fields do provide good functional advantages, performance also needs to be considered prior to using this support.

CCSID Support

CCSID (Coded Character Set Identification) enhancements support the dynamic conversion of data from one language to another. The support allows jobs, files, and fields within files to be tagged with an identification of the code page currently being used. For a more detailed description of this support refer to the AS/400 National Language Support Planning Guide.

The main effect to performance from CCSID support is from the character data conversion required when either the CCSID of the job and the file/field do not match or when either of these CCSID values is not set to 65535. The amount of additional CPU required for this conversion will vary somewhat depending on the amount of character data that needs to be converted. Since the impact of this conversion can be significant to normal database operations, users should exercise some caution when implementing this function. For example, it may be best to consider doing CCSID conversion only on fields that need the conversion done instead of all character data in the given database file.

Sort Sequence

DB2/400 sort sequence support provides application developers and end users with an easy method of producing sorted data for a particular language or culture. A set of unique and shared sort sequence tables are included on the AS/400. Developers can refer to sort sequences when creating applications using database, Query/400, RPG, COBOL, C, and ILE/C compilers, as well as SQL precompilers.

The performance of sort sequence support should be compared to the alternative methods that users have available on the AS/400. For example, users who desire to use a different sorting sequence in QUERY/400 queries can create a translation table and then specify this translation table in the ″select alternate collating sequence″ option in QUERY/400. However, comparisons of these two methods show that sort sequence support will provide a noticeable improvement in performance (ranging from 5-40%) versus using the translation table method.

Users who would like to learn more about sort sequences should refer to the *National Language Support Planning Guide*.

Variable Length Fields

Variable length field support allows a user to define any number of fields in a file as variable length, thus potentially reducing the number of bytes that need to be stored for a particular field.

Description

Variable length field support on the AS/400 has been implemented with a spill area, thus creating two possible situations: the non-spill case and the spill case. With this implementation, when the data overflows, all of the data is stored in the spill portion. An example would be a variable length field that is defined as having a maximum length of 50 bytes and an allocated length of 20 bytes. In other words, it is expected that the majority of entries in this field will be 20 bytes or less and occasionally there will be a longer entry up to 50 bytes in length. When inserting an entry that has a length of 20 bytes or less that entry will be inserted into the allocated part of the field. This is an example of a non-spill case. However, if an entry is inserted that is, for example, 35 bytes long, all 35 bytes will go into the spill area.

To create the variable length field just described, use the following SQL/400 statement:

```
CREATE TABLE library/table-name
          (field VARCHAR(50) ALLOCATE(20) NOT NULL)
```

In this particular example the field was created with the NOT NULL option. The other two options are NULL and NOT NULL WITH DEFAULT. Refer to the NULLS section in the SQL/400 Reference Guide to determine which NULLS option would be best for your use. Also, for additional information on variable length field support, refer to either the SQL/400 Reference Guide or the SQL/400 Programmer′s Guide.

Performance Expectations

- Variable length field support, when used correctly, can provide performance improvements in many environments. The savings in I/O when processing a variable length field can be significant. The biggest performance gains that will be obtained from using variable length fields are for description or comment types of fields that are converted to variable length. However, because there is additional overhead associated with accessing the spill area, it is generally not a good idea to convert a field to variable length if the majority (70-100%) of the records would have data in this area. To avoid this problem, design the variable length field(s) with the proper allocation length so that the amount of data in the spill area stays below the 60% range. This will also prevent a potential waste of space with the variable length implementation.
- Another potential savings from the use of variable length fields is in DASD space. This is particularly true in implementations where there is a large difference between the ALLOCATE and the VARCHAR attributes AND the amount of spill data is below 60%. Also, by minimizing the size of the file, the performance of operations such as CPYF (Copy File) will also be improved.

- When using a variable length field as a join field, the impact to performance for the join will depend on the number of records returned and the amount of data that spills. For a join field that contains a low percentage of spill data and which already has an index built over it that can be used in the join, a user would most likely find the performance acceptable. However, if an index must be built and/or the field contains a large amount of overflow, a performance problem will likely occur when the join is processed.
- Because of the extra processing that is required for variable length fields, it is not a good idea to convert every field in a file to variable length. This is particularly true for fields that are part of an index key. Accessing records via a variable length key field is noticeably slower than via a fixed length key field. Also, index builds over variable length fields will be noticeably slower than over fixed length fields.
- When accessing a file that contains variable length fields through a high-level language such as COBOL, the variable that the field is read into must be defined as variable or of a varying length. If this is not done, the data that is read in to the fixed length variable will be treated as fixed length. If the variable is defined as PIC X(40) and only 25 bytes of data is read in, the remaining 15 bytes will be space filled. The value in that variable will now contain 40 bytes. The following COBOL example shows how to declare the receiving variable as a variable length variable:

  ```
  01 DESCR.
     49 DESCR-LEN      PIC S9(4) COMP-4.
     49 DESCRIPTION    PIC X(40).

  EXEC SQL
     FETCH C1 INTO DESCR
  END-EXEC.
  ```

  For more detail about the vary-length character string, refer to the SQL/400 Programmer′s Guide.

  The above point is also true when using a high-level language to insert values into a variable length field. The variable that contains the value to be inserted must be declared as variable or varying. A PL/I example follows:

  ```
  DCL FLD1 CHAR(40) VARYING;
  FLD1 = XYZ Company;

  EXEC SQL
     INSERT INTO library/file VALUES
        (″001453″, FLD1, ...);
  ```

  Having defined FLD1 as VARYING will, for this example, insert a data string of 11 bytes into the field corresponding with FLD1 in this file. If variable FLD1 had not been defined as VARYING, a data string of 40 bytes would be inserted into the corresponding field. For additional information on the VARYING attribute, refer to the PL/I User′s Guide and Reference.
- In summary, the proper implementation and use of DB2/400 variable length field support can help provide overall improvements in both function and performance for certain types of database files. However, the amount of

improvement can be greatly impacted if the new support is not used correctly, so users need to take care when implementing this function.

Reuse Deleted Record Space

Description of Function

This section discusses the support for reuse of deleted record space. This database support provides the customer a way of placing newly-added records into previously deleted record spaces in physical files. This function should reduce the requirement for periodic physical file reorganizations to reclaim deleted record space. File reorganization can be a very time consuming process depending on the size of the file and the number of indexes over it. To activate the reuse function, set the Reuse deleted records (REUSEDLT) parameter to *YES on the CRTPF (Create Physical File) or CHGPF (Change Physical File) commands. The default value when creating a file is *NO (do not re-use).

Comparison to Normal Inserts

Inserts into deleted record spaces are handled differently than normal inserts and have different performance characteristics. For normal inserts into a physical file, the database support will find the end of the file and seize it once for exclusive use for the subsequent adds. Added records will be written in blocks at the end of the file. The size of the blocks written will be determined by the default block size or by the size specified using an Over-ride Database File (OVRDBF) command. The SEQ(*YES number of records) parameter can be used to set the block size.

In contrast, when re-use is active, the database support will process the added record more like an update operation than an add operation. The database support will maintain a bit map to keep track of deleted records and to provide fast access to them. Before a record can be added, the database support must use the bit-map to find the next available deleted record space, read the page containing the deleted record entry into storage, and seize the deleted record to allow replacement with the added record. Lastly, the added records are blocked as much as permissible and then written to the file.

To summarize, additional CPU processing will be required when re-use is active to find the deleted records, perform record level seizes and maintain the bit-map of deleted records. Also, there may be some additional disk IO required to read in the deleted records prior to updating them. However, this extra overhead is generally less than the overhead associated with a sequential update operation.

Performance Expectations

The impact to performance from implementing the reuse deleted records function will vary depending on the type of operation being done. Following is a summary of how this function will affect performance for various scenarios:

- When blocking was not specified, re-use was slightly faster or equivalent to the normal insert application. This is due to the fact that reuse by default blocks up records for disk IOs as much as possible.
- Increasing the number of indexes over a file will cause degradation for all insert operations, regardless of whether reuse is used or not. However, with reuse activated, the degradation to insert operations from each additional index is generally higher than for normal inserts.
- The RGZPFM (Reorganize Physical File Member) command can run for a long period of time, depending on the number of records in the file and the number of indexes over the file. Even though activating the reuse function may cause some performance degradation, it may be justified when considering reorganization costs to reclaim deleted record space.
- The reuse function can always be de-activated if the customer encounters a critical time window where no degradation is permissible. The cost of activating/de-activating reuse is relatively low in most cases.
- Because the reuse function can lead to smaller sized files, the performance of some applications may actually improve, especially in cases where sequential non-keyed processing of a large portion of the file(s) is taking place.

DB2/SMP Feature

Introduction

The symmetrical multiprocessing (SMP) feature provides additional query optimization algorithms for retrieving data. In addition, the DB2/SMP feature provides application transparent support for parallel query operations on a single tightly-coupled multi-processor AS/400 system (shared memory and disk). The database manager can automatically activate parallel query processing in order to engage one or more system processors to work simultaneously on a single query. The response time can be dramatically improved when a processor bound query is executed in parallel on multiple processors. The purpose of this section is to:

- Introduce new query optimization algorithms available with the DB2/SMP feature
- Briefly discuss decision support (DSS) queries which will realize the most benefit with the SMP feature.
- Provide guidance to help estimate DSS query capacity on various AS/400 systems.

New Query Optimization Algorithms

The DB2/SMP feature provides the following new query optimization algorithms:

- Parallel table scan

  Provides parallel operations for queries requiring a sequential scan of the entire table. Multiple tasks are used to scan the same table concurrently. Each task will perform selection and column processing on a table partition and return selected records to the requester. The response time improvement for a parallel table scan scales closely to the number of processors participating. For example, the response time for a table scan can be up to 4 times faster when run in parallel on a 4-way processor.

- Index only access (parallel and non-parallel)

  Provides performance improvement by extracting a query answer from an index rather than performing random I/Os against a physical table. For this to happen, all columns that are referenced in a query must exist within an index. Response time improvements can be up to 5 times faster for some queries.

- Parallel key selection

  Provides parallel index operations for key selection. Multiple tasks are used to scan the same index concurrently. Each task will search a different key range and selected records are returned to the requester.

- Hashing algorithms

  Provides an optimization alternative for group by and some join queries. This method avoids having to utilize an index and therefore avoids having to perform random I/Os to retrieve the results. Instead, a temporary partitioned hash table can be used. This table can be processed by large and efficient sequential I/Os and often utilizing parallel table scan to provide the results. Response time improvements for group by queries can be up 6 times better and some joins can be up to 25 times improved (4 to 10 times is more typical).

The SMP feature was available V3R1 on AS/400 IMPI models and became available V3R7 for AS/400 RISC models. For more information on the SMP feature and the new algorithms, see *TNL SN41-3680 to SC41-3611-00*.

Decision Support Queries

The SMP feature is most useful when running decision support (DSS) queries. DSS queries generally give answers to critical business questions tend to have the following characteristics:

- examine large volumes of data
- are far more complex than most OLTP transactions
- are highly CPU intensive
- includes multiple order joins, summarizations and groupings

DSS queries tend to be long running and can utilize much of the system resources such as processor capacity (CPU) and disk. For example, it is not unusual for DSS queries to have a response time longer than 20 seconds. In fact, complex DSS queries may run an hour or longer. The CPU required to run a DSS query can easily be 100 times greater than the CPU required for a typical OLTP transaction. Thus, it is very important to choose the right AS/400 system for your DSS query and data warehousing needs.

SMP Performance Summary

The SMP feature provides performance improvement for query response times. The overall response time for a set of DSS queries run serially at a single work station may improve 25 to 58 percent when SMP support is enabled. The amount of improvement will depend in part on the number of processors participating in each query execution and the optimization algorithms used to implement the query. Some individual queries can see significantly larger gains. Queries that are able to utilize the new hash join algorithm may see up to a 25 times improvement in query response time. In addition, query throughput may improve 18 to 25 percent because the new optimization algorithms require less CPU resource. The new hashing algorithms also dramatically reduce the number of disk IOs.

Capacity Planning

The Capacity Planning sections contain the following information:

- Initial system sizing recommendations for data warehouses
- Detailed capacity planning information for various AS/400 models. This information will be useful when you are able to determine a customer′s average DSS query response time and want to compare running a query workload on other AS/400 models or with the SMP feature enabled.
- Capacity planning tips

System Sizing Recommendations for Data Warehouses

The following table gives some high-level guidance for choosing the AS/400 system for Data Warehouses based on the size of the database and/or the maximum number of concurrent users.

Table 16. System Sizing Recommendations for AS/400 Data Warehouses

System	Maximum Data in Gigabytes	Maximum number of concurrent users
40S	15	15
50S 2120	220	20
50S 2121	220	30
53S 2154	350	40
53S 2155	350	65
53S 2156	350	125

Note:

1. The maximum amount of main storage exists on each system.
2. Query workloads are assumed to be comprised of the following query mixture:
   - Simple queries (no joins or group by aggregation) 80%
   - Medium queries (2-way joins, group by aggregation) 15%
   - Complex queries (Unions, subselects) 5%
3. Simple query workloads may also include the use of any Multi-Dimensional Database product.
4. If your database size is greater than 350 Gigabytes or the number of concurrent users is greater than 125, you will require a multisystem implementation which may include the the DB2 Multisystem feature. DB2 Multisystem provides the capability of horizontally partitioning a table across multiple systems and running a single query in parallel. Performance is improved due to parallel operations and because of the table partitioning. Each system need only scan a fraction of of the entire database when a query is run.

Capacity Planning based upon Average Query Response Time

This section provides some guidance to help you estimate DSS query capacity on various AS/400 systems, with and without the SMP feature enabled. The chart was developed based upon studying the results of various customer and synthetic DSS workloads. The workloads contained various sized files ranging from 25 records up to 100 million records. A broad range of DSS support queries, from simple to complex were measured. Queries that utilized joins, group by, and summarizations were commonplace. The database structure, the index structure, and the query syntax are all assumed to be optimal. The SMP numbers in the the chart show a range of performance based upon an estimate of:

- the percentage of the DSS queries that might be helped by SMP
- the query benefit provided by parallelism

How to use the chart

Calculating the capacity for DSS query workloads can be difficult due to vast variability of the queries. The capacity chart uses an average query response time that might be observed in a customer environment over a long period of time such as a day. Obviously, during this time period there will be great variance in terms of complexity of the queries, the size of the tables queried, the response times of the individual queries, and the load put on the AS/400 system.

For the capacity chart, we have used 180 seconds to represent a customer′s average query time. An average of 180 seconds would indicate a majority of simple DSS queries being executed during the time window. If this does not accurately reflect your customer′s environment, you can estimate new average response times and system capacities by performing the calculations that follow the capacity chart:

Table 17. System Capacity Planning - DSS Queries

Model	CPUs	Avg Response Time in Seconds		Capacity in Queries/hour	
		No SMP	SMP range	No SMP	SMP range
30S 2411	1	180	-	40	-
30S 2412	2	173	93-130	86	105-114
E95	4	188	78-117	148	179-195
F97 & 320	4	154	64-96	225	272-297
50S 2121	1	100	-	98	-
53S 2154	1	86	-	159	-
53S 2155	2	86	46-64	244	296-324
53S 2156	4	86	36-54	372	451-493

Note:

1. The average CPU reduction when SMP is enabled 18-25%.
2. Capacity numbers are based on 100% CPU utilization and assume that the system is dedicated to query processing.
3. Information in chart based on assumptions listed in the next section.

You can estimate new average response times and system capacities by performing the following calculations against the values in the capacity chart:

- Determine the customers average response time
- Compute the following query response time ratio:

  ratio = customer′s average response time/average response time from table

- Multiply all response times by the ratio to get new response times
- Divide all capacities by the ratio to get new capacities
- For example, if the customer′s average response time is 38.5 seconds on an F97 processor, to calculate the new F97 values, perform the following calculations:

```
Current F97 row:  F97 & 320  4   154   64 96     225   272 297

                  Ratio = 38.5/154 = .25

                  Response times      Capacities

                  154 (*.25) = 38.5     225/.25 =  900   No SMP
                   64 (*.25) = 16.0     272/.25 = 1088   SMP
                   96 (*.25) = 24.0     297/.25 = 1188   SMP

New F97 row:      F97 & 320  4   38.5  16 24      900   1088 1188
New 53S 2156 row: 53S 2156   4   21.5   9 13.5  1488   1804 1972
```

Capacity Planning Assumptions

The following assumptions were used to help generate the capacity chart:

1. DSS query workloads can be characterized by an average response time. The average response time will increase as the size of the customer′s database size increases.
2. Given all of a customer′s DSS queries, typically 50%-70% of the queries will utilize the SMP support.
3. For queries that utilize SMP, the response time will scale relative to the number of CPUs. The scaling range equals 1 to 1.5 times the number of CPUs involved in the query execution. For example, on a 4-way CPU system, response time will be 1/4 to 1/6 the time compared to executing the query on just one of the processors.
4. Hash group by/join algorithms will be utilized in about 70% of all queries that can utilize the SMP support. About 50-75% of the IOs will be eliminated when the new hashing algorithms are used.
5. Table scans will be utilized in about 10% of all the DSS queries.
6. For SMP queries, CPU consumption will decrease up to 35% due to the new optimization algorithms and because of the reduction in disk IOs.
7. A DSS query workload will utilize at least 50% of the system processor capacity when run on an AS/400 30S 2411.

Capacity Planning Tips

Here are some suggestions that may improve your DSS query performance when utilizing the SMP support:

- Add additional memory. 20%-25% of the active database should reside in main memory.

- Add additional disks and limit the the number of disks per controller to 8 if possible. This is especially true if you are using 9337 DASD and the 6501 DASD IOP. as this will cause more efficient use of active memory.

- Utilize fast DASD (6503,6506,and 6507) and DASD IOPs (6502 and 6512). Spread the IOPs evenly among the system busses.

- Ensure that the database is spread evenly over multiple DASD arms. Installing DASD that is all the same size helps ensure even spreading.

- For smaller database sizes (< 30GB), you should have 2-3 DASD arms per CPU to get good performance.

- Utilize RISC hardware. RISC systems have faster system busses with larger bandwidths than those found on IMPI systems. In addition, disk IO sizes are larger which will results in fewer disk IOs.

- Be sure that there is enough space on the the system auxiliary storage pool (ASP) to allow the database manager to create temporary files for query execution. Do not exceed 70% capacity on the system ASP.
- Under a heavy system load, limit the amount of query parallelism. The degree of parallel activity can be controlled by the user via the CHGSYSVAL (parm QQRYDEGREE) and CHGQRYA (parm Degree) CL commands.

DB2 Multisystem for OS/400

DB2 Multisystem for OS/400 offers customers the ability to distribute large databases across multiple AS/400s in order to gain nearly unlimited scalability and improved performance for many large query operations. The multiple AS/400s are coupled together in a shared-nothing cluster where each system uses its own main memory and disk storage. Once a database is properly partitioned among the multiple nodes in the cluster, access to the database files is seamless and transparent to the applications and users that reference the database. To the users, the partitioned files still behave as though they were local to their system.

This section will provide information on what level of performance improvements to expect from DB2 Multisystem as well as tips and techniques on how to install and use this product for optimal performance. However, this section should not be viewed as a complete guide to performance for DB2 Multisystem. It is recommended that in addition to the information provided here, you should obtain the following documents to help understand more about both the key performance and functional aspects of this product.

- *DB2 Multisystem for OS/400*, SC41-3705-00

  This document is an excellent overall reference for this product and contains several aspects of performance that will not be covered in this document, in particular some items on distributed query optimization and processing.

- *Slash DB2/400 Query Time with Parallel Processing*

  This article (found in the April 1996 edition of the NEWS/400 magazine) helps explain key performance and functional concepts of DB2 Multisystem.

These documents and the information in this section assumes that you are familiar with nondistributed query performance on the AS/400 and that you have a good overall background in database concepts. Other documents that can help you with this information include:

- *DB2 for OS/400 SQL Reference*
- *DB2 for OS/400 SQL Programming*
- *DB2 for OS/400 Database Programming*
- *CL Reference Guide*

Planning for DB2 Multisystem

The most important aspect of obtaining optimal performance with DB2 Multisystem is to plan ahead for what data should be partitioned and how it should be partitioned. The main idea behind this planning is to ensure that the systems in the cluster run in parallel with each other as much as possible when processing distributed queries while keeping the amount of communications data traffic to a minimum. Following is a list of items to consider when planning for the use of distributed data via DB2 Multisystem.

- Avoid large amounts of data movement between systems. A distributed query often achieves optimal performance when it is able to divide the query among several nodes, with each node running its portion of the query on data that is local to that system and with a minimum number of accesses to remote data on other systems. Also, if a file that is heavily used for transaction processing is to be distributed, it should be done such that most of the database accesses are local since remote accesses may add significantly to response times.
- Choosing which files to partition is important. The largest improvements will be for queries on large files. Files that are primarily used for transaction processing and not much query processing are generally not good candidates for partitioning. Also, partitioning files with only a small number of records will generally not result in much improvement and may actually degrade performance due to the added communications overhead.
- Choose a partitioning key that has many different values. This will help ensure a more even distribution of the data across the multiple nodes. In addition, performance will be best if the partitioning key is a single field that is a simple data type.
- It is best to choose a partition key that consists of a field or fields whose values are not updated. Updates on partition keys are only allowed if the change to the field(s) in the key will not cause that record to be partitioned to a different node.
- If joins are often performed on multiple files using a single field, use that field as the partitioning key for those files. Also, the fields used for join processing should be of the same data type.
- It will be helpful to partition the database files based on how quickly each node can process its portion of the data when running distributed queries. For example, it may be better to place a larger amount of data on a large multiprocessor system than on a smaller single processor system. In addition, current normal utilization levels of other resources such as main memory, DASD and IOPs should be considered on each system in order to ensure that no one individual system becomes a bottleneck for distributed query performance. For information on how to customize your database partitioning, refer to the ″DB2 Multisystem for OS/400″ document mentioned above.
- For the best query performance involving distributed files, avoid the use of commitment control when possible. DB2 Multisystem uses two-phase commit, which can add a significant amount of overhead when running distributed queries.

In addition to these items, the document and article referenced above contain other key concepts that should be considered while planning your data distribution via DB2 Multisystem.

Performance During Data Distribution

Generally, partitioning large database files across multiple systems can be a long process during which the data in the files is unavailable. Following is a list of items that should be considered prior to actually partitioning the files to help reduce the time this process may take.

- The use of Opticonnect will result in significantly better distribution times than using other alternatives such as a 16Mbps Token Ring LAN. Opticonnect will also help improve performance for distributed queries that result in large amounts of data being moved from node to node to complete the query.

- There are basically two recommended methods of distributing data from a local system to a set of systems linked together with DB2 Multisystem. One method is to use the Change Physical File (CHGPF) command with the NODGRP and PTNKEY parameters. This command will need to be issued against each database file to be distributed. Any existing logical files for this file will also be rebuilt on a per node basis. The second method is to create a new physical file with the same data format as the original and with the node group and partition key specified (this can be done either via the Create Physical File (CRTPF) command or the SQL CREATE TABLE command), and then issue a Copy File (CPYF) command to copy the data from the original file to the new distributed file. Measurement results show that the performance of these methods is about equal.

  Note that there is a faster and slower version of both the CHGPF and CPYF operations for distributing files. The faster version sends large buffers of records at a time while the slower version sends one record at a time. To see if the fast version is being done, look for occurrences of the CPC9203 message in the joblog of the job doing the distribution, stating how many records were copied to each node. If these messages do not appear, the slower version is being used. The factors that influence which version is used are listed in more detail in the DB2 Multisystem for OS/400 document mentioned above.

- To help the distribution process, it may be best to keep the number of logical files to a minimum for the physical files that are being distributed. These logical files can then be built via the Create Logical File (CRTLF) or the SQL CREATE INDEX command at a later time, possibly in background batch jobs. This approach is generally faster than having the system maintain or build the indexes on each node as the physical data is distributed. However, you will have to issue the index builds separately and they will tend to cause high CPU utilization while they are occurring, so this must be considered as well. If you need certain key indexes to exist as soon as the data distribution is done, you should let the CPYF or CHGPF operations handle these for you.

- It will be to your benefit to avoid the use of commitment control or journaling while distributing database files. The use of these options will add significantly to the overall distribution time.

- The time for data distribution may also be helped by having several jobs running at the same time, each distributing a different file. Although this is best accomplished where the system doing the distribution is a multiprocessor system, this can also apply to single processor systems. The key to making this work is to avoid a bottleneck on a resource such as main memory, DASD, CPU or the communications lines or IOPs. It may be best to try this by adding one job at a time and monitoring system performance to see if any resources are becoming overutilized.

Distributed Query Performance

The performance of queries run over distributed data will in many cases improve significantly compared to the performance that had been achieved running these same queries on a single system. However, there also may be queries that show little or no performance gain, with some possibly showing degradation in performance. The following information should help determine what level of performance to expect when running queries over distributed data. Again, it is important to reference the above mentioned documents in conjunction with the information provided here in order to gain a more complete picture of distributed query performance.

- Use of the new ASYNCJ parameter on the Change Query Attributes (CHGQRYA) command is very important to achieving the best performance levels for distributed queries. The value specified for this parameter will greatly affect the response time for distributed queries by altering the degree of parallelism allowed as well as the amount of work done by the temporary result writer jobs. Note that this command needs to be issued on a per job basis as there is no global system level value that can be changed. For more information on the use of this command for distributed queries, refer to the *DB2 Multisystem for OS/400* document.
- There is now a distributed query optimizer that operates only on distributed queries. This optimizer determines what steps are necessary to efficiently run the distributed query and what nodes will process these individual steps. Local level optimization on each node is still handled by the previously existing query optimizer.
- The use of Opticonnect is recommended for the best overall performance of distributed queries. Although good planning will minimize the amount of communications overhead needed for many distributed queries, there still will be a fair amount of cross-system data traffic in many DB2 Multisystem environments. Using Opticonnect will result in noticeably better response times for queries with a significant amount of cross-system data movement and will in general help reduce the communications overhead for users of DB2 Multisystem.
- Generally, the best performance gains from DB2 Multisystem will be for queries that exhibit the following characteristics:
  - The query processes a large number of records
  - The query can be divided such that subsets of the records it processes can be queried on multiple nodes in parallel
  - Each part of the divided query returns a small number of records to the coordinating system where the query originated

  For queries that meet these criteria, performance can be expected to improve in nearly a linear progression with the number of systems involved in running the query. In addition, if any of the systems used are multiprocessor systems, the improvement on these nodes may also be multiplied by the enhancements provided by DB2 Symmetric Multiprocessing for OS/400 (SMP). For example, a query that had previously been run on a single processor and is now being run on three four-way systems could experience a run time that is one-twelfth of what it had been. Although this amount of improvement may not be realized in most queries, there will still be many queries that will experience large improvements in performance.
- Queries that read and process a small number of records may experience some level of performance improvement when running over distributed data,

but the percentage of improvement will in many cases be much less than queries over large files. For queries of this type, the amount of improvement will often be a factor of the speed of the connection between the systems, and in some cases, this may cause the query to run longer than it had on a single system.

- Queries that read and process a large number of records but that also return a large number of records to the coordinating system will in many instances not experience the almost linear improvements mentioned above. In this case, the individual nodes may still be able to process a subset of the records efficiently, but the response time may be affected by how quickly the records in the individual answer sets can be transferred back to the coordinator and how quickly this system can receive and process them as well.
- The performance of join queries on distributed data is closely linked to how much data needs to be transferred between nodes to perform the join. The best performing join queries are where all of the corresponding records of the files being joined exist on the same node so that no data is moved to other nodes to perform the join. These types of joins should improve nearly linearly with the number of nodes, although this again depends on the amount of data that needs to be transferred back to the coordinating system and the additional processing that will be needed there. Other join operations that need to move data between nodes to do the join will vary widely in how much improvement is achieved, and in some cases, may end up with a significant degradation. For this reason, partitioning of commonly joined files needs to be planned such that the most common join operations end up moving only smaller amounts or no data between nodes. For a more detailed discussion on distributed join performance, refer to Chapter 6 of the previously mentioned DB2 Multisystem for OS/400 document.
- Queries that specify selection criteria on a single file may end up doing all the processing of that query on a single node if the optimizer determines that all the records matching the criteria exist on that node. In this case, the amount of performance improvement for this type of query will vary depending on how quickly the system at that node can process the query and return the results to the coordinating system. However, there are certain restrictions that a query must meet in order to be directed to a single particular node. More information on this type of query can be found in Chapter 6 of the DB2 Multisystem document.
- For most distributed queries and in particular for queries involving ordering of data, it is best to specify the ALWCPYDTA(*OPTIMIZE) parameter on the Open Query File (OPNQRYF) and Start SQL (STRSQL) commands and also on the Create SQLxxx (CRTSQLxxx) commands. This option allows the optimizer the most flexibility in choosing what method to use (an index or a sort) to order the records on each node.
- To achieve the fastest retrieval of data from a distributed file, you can issue the Override Database File (OVRDBF) command with the DSTDTA(*BUFFERED) parameter specified. For more information on this option, refer to Chapter 5 of the previously mentioned DB2 Multisystem for OS/400 document.

In addition to the above information, there are many other items to consider to understand distributed query optimization and how to obtain optimal performance levels when using DB2 Multisystem support. The following items (as well as many of the above) are covered in the above mentioned DB2 for Multisystem document.

- ORDER BY and GROUP BY operations
- Reusable and non-reusable ODPs
- Temporary result writers (new for DB2 Multisystem)
- Optimizer messages
- Changes to the Change Query Attributes (CHGQRYA) command

Chapter 5. Communications Performance

There are many factors that affect the performance of an AS/400 in a communications environment. This section discusses some of the common factors and offers guidance on how to achieve the best possible performance.

CPB and NetPerf Benchmark Descriptions

CPB (Communications Performance Benchmark) and NetPerf are benchmarks that consist of C programs that run on the AS/400. A program running on a client AS/400 requests data from a program running on a server AS/400. Multiple instances of CPB or NetPerf can be executed over multiple connections to increase the system load. The programs communicate with each other with one of the following programming interfaces:

1. Sockets
2. ICF
3. CPI-C

Whereas most ′real′ application programs will process data in some fashion, these benchmarks merely copy and transfer the data from memory. Therefore, additional consideration must be given to account for other normal application processing costs (for example, higher CPU utilization and higher response times due to database accesses).

To demonstrate communications performance in several different ways, several scenarios were measured and analyzed:

1. A large transfer with the server AS/400 system **sending each record repetitively** using the default blocking provided by OS/400 to the client AS/400 system.
2. A large transfer with the server AS/400 system **flushing the communications buffer after each record** (FRCDTA keyword for ICF) to the client AS/400 system. That is, each record will be forced out of the server system to the client system without waiting to be blocked with any subsequent data. Note that ICF and CPI-C support blocking, Sockets does not.
3. A large transfer with the server AS/400 system **sending each record requiring a synchronous confirm** (e.g., CONFIRM keyword for ICF) to the client AS/400 system. That is, each record is forced out of the server system to the client system. The server system program then waits for the client system to respond with a confirm (acknowledgement). The server application cannot send the next record until the confirm has been received.
4. A **send/receive** scenario where the client and server send a record back and forth (serially).

5.1 WAN Protocols, Lines, and IOPs

This section discusses the performance characteristics of Wide Area Network (WAN) protocols, lines, and IOPs. The CPB benchmark was used to illustrate the difference between protocols, the effectiveness of the line capacity, and the effect of frame size variation.

Table 18. Large Transfer with Wide Area Networks

Large Transfer with Wide Area Networks Application Program Transfer (CPB Benchmark) AS/400 50s-2122 --> 50s-2122 V4R1				
Protocol	**Line Speed (kbps)**	**Frame Size (bytes)**	**Data Rate (kbps)**	**CPU Utilization (%)**
APPC/SDLC	64	265	58	0.4/0.8
APPC/SDLC	64	2057	62	0.1/0.2
APPC/X.25	64	256	60	0.1/0.2
APPC/X.25	64	4096	62	0.1/0.2
BSC	64	2048	63	0.3/0.6
ASYNC	28.8	2048	23	0.2/0.2
TCP/SLIP	115.2	4096	91	1.2/1.6
TCP/FRelay	2048	1590	1927	5.7/8.4

Note:
- The performance of V4R2 and V4R3 should be similar
- repetitive 2048-byte sends
- 2629 Communications IOP
- CPU utilizaton is listed as ′source data / target data′
- FRelay did 32763-byte repetitive sends instead

WAN Performance Tips and Techniques

WAN Line Performance Considerations:

- In many cases, the communications line is the largest contributor to overall response time. Therefore, it is important to closely plan and manage its performance. In general, having the appropriate line speed is the most key consideration for having best performance.
- A common misconception exists in sizing systems with communications. Many incorrectly believe that each attached line consumes CPU resource in a uniform fashion, and therefore, exact statements can be made about the number of lines that any given AS/400 model can support. For example, if the sales pages say that a particular AS/400 model supports 64 lines, it does not mean that any given customer can run their workload fully utilizing those 64 lines. It is merely a rough guideline stating the suggested maximum for that model (in some cases, it is the maximum configuration possible).
- Communications applications consume CPU and IOP resource (to process data, to support disk I/O, etc.) and communications line resource (to send and receive data or display I/O). The amount of line resource that is consumed is proportional to the total number of bytes sent or received on the line. Some additional CPU resource is consumed to process the communications software to support the individual sends (puts or writes) and receives (gets or reads). Communications IOP resource is also consumed to support the line activity.

  So the best question to ask is NOT ″How many lines does my system support?″, but rather, ″How many lines does my workload require, and what AS/400 model is required to accommodate this load?″.

- To estimate the utilization of a half duplex line:
  utilization = (bytes in + bytes out) * 800 / time / linespeed
  where time = total # of seconds
  and linespeed = the speed of the line in bits per second

- For a full duplex line (e.g., X.25, ISDN), the AS/400 Performance Tools report utilization as follows:
  utilization = (bytes in + bytes out) * 400 / time / linespeed

  For example, if the send direction is 100% busy and the receive direction is 0% busy, the Performance Tools will report an overall 50% line utilization.

- Note from the data in Table 18 on page 80, that the system is able to drive the line to a high utilization. The difference of the data rate and the line speed is due to the overhead of header bytes, and line turn around ′dead′ time.

- When several sessions use a line concurrently, the aggregate data rate may be higher. This is due to the inherent inefficiency of a single session in using the link. In other words, when a single job is executing disk operations or doing non-overlapped CPU processing, the communications link is idle. If several sessions transfer concurrently, then the jobs may be more interleaved and make better use of the communications link.

- For interactive environments, keeping line utilization below 30% is recommended to maintain predictable and consistent response times. Exceeding 50% line utilization will usually cause unacceptable response times. The line utilization can be measured with the AS/400 performance tools.

- For large transfer environments, or for environments where only a small number of users are sharing a line, having a higher line utilization may yield acceptable response times. In fact, maximizing line utilization means maximizing throughput for that single job.

- For large transfers, use large frame sizes for best performance. Fewer frames make more efficient use of the CPU, the IOP, and the communications line (higher effective data rate).

- To take advantage of these large frame sizes, they must be configured correctly. The MAXFRAME parameter on the LIND must reflect the maximum value. For X.25, the DFTPKTSIZE and MAXFRAME must be increased to its maximum value. Also, go to the APPC and TCP section to ensure other related parameters are optimized.

- Configuring a WAN line as full-duplex may provide a higher throughput for certain applications that can take advantage of that, or for multiple-user scenarios.

- In general, the physical interface does not noticeably affect performance for a given protocol assuming that all other factors are held constant (e.g., equal line speeds). For example, if SDLC is used with a line speed of 19.2 kbps, it would not matter if a V.35, RS232, or an X.21 interface was used (all other factors held constant).

- For SDLC environments, polling is an important consideration. For a discussion on this, refer to the polling section in the performance chapter of the *Communications: Management Guide, SC41-0024.* In summary, parameters can be adjusted to change the rate at which a line is polled. Polls consist of small frames sent across the line and are processed by the IOPs. Therefore, polling contributes to line utilization and IOP utilization.

- The CPU usage (i.e., CPU time per unit of data) for fractional T1 support and other high speed WAN connections is similar to any other line running the same type of work. As the speed of a line increases from a traditional low speed to a high speed (e.g., 640 kbps), performance characteristics may change.
  - Interactive transactions may be slightly faster
  - Large transfers may be significantly faster
  - A single job may be too serialized to utilize the entire bandwidth
  - High throughput is more sensitive to frame size
  - High throughput is more sensitive to application efficiency
  - System utilization from other work has more impact on throughput

WAN IOP Performance Considerations

- For the data listed in Table 18 on page 80, the IOPs handled the load with a very low IOP utilization (not provided in the table). When using the new IOPs provided with V4R1 (or more recent IOPs), they typically won′t be the bottleneck in WAN environments.
- For interactive environments it is recommended not to exceed 60% utilization on the communications IOP. Exceeding this threshold in a large transfer environment or with a small number of concurrent users may still offer acceptable performance. Use the AS/400 performance tools to get utilization.
- Even though an IOP can support certain configurations, a given AS/400 model may not have enough system resource (for example, CPU processing capacity) to support the workload over the lines.
- In communications environments where errors are common, the use of smaller frame sizes may offer better performance by limiting the size of the re-transmissions. Having errors may also impact the number of communications lines that can run concurrently.
- The values for IOP utilization in SDLC environments do not necessarily increase consistently with the number of work stations or with the amount of workload. This is because an IOP can spend more time polling when the application is not using the line. Therefore, it is possible to see a relatively high IOP utilization at low throughput levels.

These guidelines are based on a particular set of test cases. Results in other environments may vary significantly.

5.2 LAN Protocols, Lines, and IOPs

This section discusses the performance characteristics of local area network (LAN) protocols, lines, and IOPs.

Table 19 on page 83 includes the data from the CPB benchmark that illustrates the difference between protocols, the effectiveness of the line capacity, and the effect of frame size variation.

Table 19. Large Transfer with Local Area Networks

Large Transfer with Local Area Networks **Application Program Transfer (CPB Benchmark)** **AS/400 50s-2122 (server) --> 50s-2122 (client)** **V4R2 (V4R3 should look similar)**				
Protocol	**Line Speed (Mbps)**	**Frame Size (bytes)**	**Data Rate (Mbps)**	**CPU Utilization (%)**
TCP/TRLAN	16	265	3.5	15.2/19.3
TCP/TRLAN	16	1024	14.4	17.6/21.1
TCP/TRLAN	16	1994	15.0	15.8/18.8
TCP/TRLAN	16	4060	15.2	15.7/21.4
TCP/TRLAN	16	8156	15.6	14.0/18.1
TCP/TRLAN	16	16393	15.7	9.7/14.9
APPC/TRLAN	16	265	6.1	2.3/5.4
APPC/TRLAN	16	1024	14.7	5.9/9.8
APPC/TRLAN	16	1994	15.2	6.1/9.4
APPC/TRLAN	16	4060	15.6	6.4/9.1
APPC/TRLAN	16	8156	15.7	6.4/9.3
APPC/TRLAN	16	16393	15.7	6.0/7.1
TCP/ETH	10	1496	9.4	13.6/16.4
TCP/ETH	100	1496	38.5	35.7/35.0
APPC/ETH	10	1496	9.6	3.9/6.3
APPC/ETH	100	1496	30.5	12.1/19.8
ATM/TR16	155	16393	65.5	40/61.4
ATM/ETH	155	1496	38.5	31.7/32.8

Note:

- CPU utilization is listed as ′server / client′ for TCP/IP
- CPU utilization is listed as ′source / target′ for APPC
- Repetitive sends of 32763 bytes each
- 2629 IOP with the TRLAN 6149 IOA
- 2810 IOP with the Eth 2838 IOA
- 2810 IOP with the ATM 2816 IOA

Figure 18 on page 84 shows the maximum data rates for each LAN IOP. The CPB benchmark was used (repetitive 32763-byte sends) with a single job transferring between two AS/400s. The IOPs on each AS/400 were varied to show the differences.

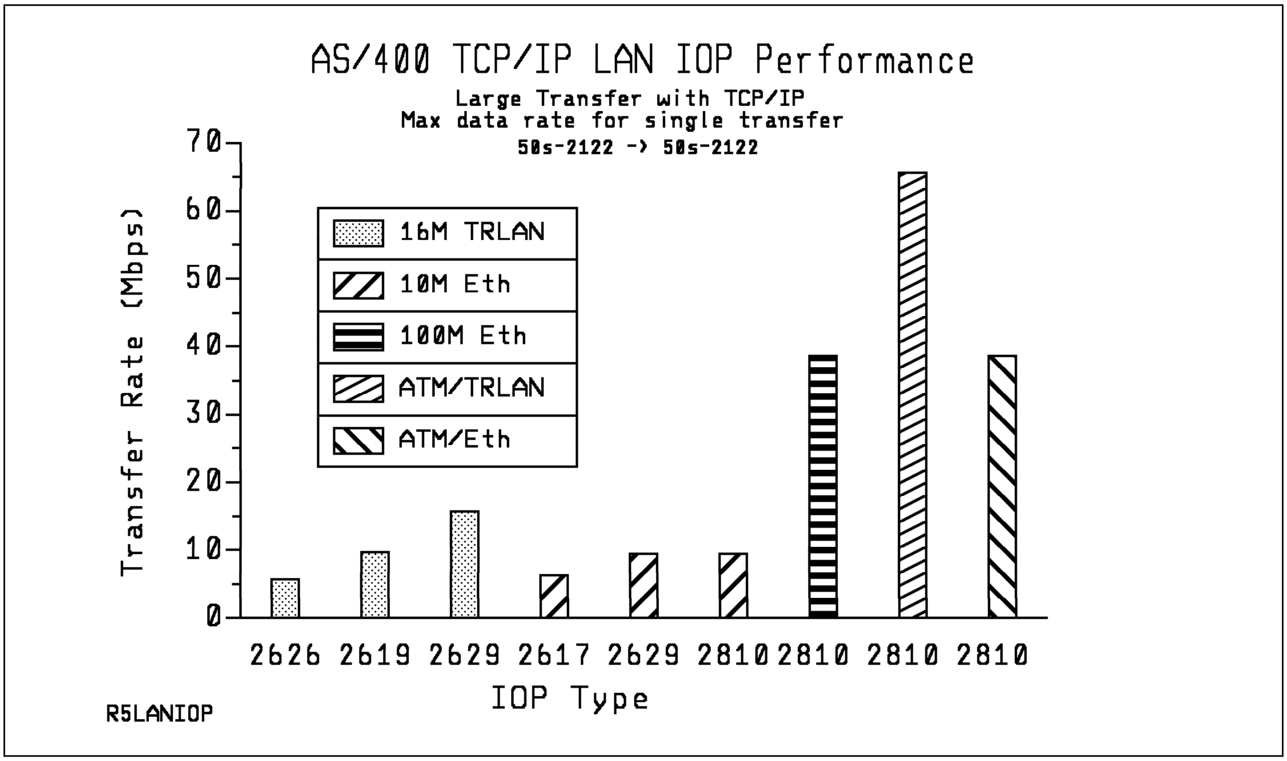


Figure 18. LAN IOP Maximum Data Rates

LAN Tips and Techniques

- No single station can or is expected to use the full bandwidth of the LAN media. It offers up to the media′s rated speed of aggregate capacity for the attached stations to share. The CPU is usually the limiting resource. The data rate is governed primarily by the application efficiency attributes (for example, amount of disk accesses, amount of CPU processing of data, application blocking factors, etc.).
- LAN can achieve a significantly higher data rate than any other supported protocol. This is due to the desirable combination of having a high media speed along with large frame sizes.
- When several sessions use a line or a LAN concurrently, the aggregate data rate may be higher. This is due to the inherent inefficiency of a single session in using the link.
- TCP/IP adaptive packet training was added beginning in V4R2. This enables the software to modify internal wait times to balance CPU utilization and round trip send/receive times. For example, the round trip time for a small token ring frame dropped from about 14 milli-seconds to about 3 ms while keeping the same CPU utilization per frame.
- In order to achieve good performance in a multi-user interactive LAN environment it is recommended to manage the number of active users so that LAN media utilization does not exceed 50% (25% for Ethernet environments because of media collisions resulting in thrashing). Operating at higher utilizations may cause poor response time due to excess queueing time for the line. In a large transfer environment where there is a small

number of users contending for the line, at any given time a higher line utilization may still offer acceptable performance.

- There are several parameters in the line description and the controller description that play an important performance role.
  - **MAXFRAME** on the line description (LIND) and the controller description (CTLD): Maximizing the frame size in a LAN environment is very important and supplies best performance for large transfers. Having configured a large frame size does not negatively impact performance for small transfers. Note that both the AS/400 system and the other link station must be configured for large frames. Otherwise, the smaller of the two maximum frame size values is used in transferring data. Bridges may also limit the maximum frame size. Note that the maximum frame size allowed is 16393 for TRLAN. Note that the default value is 1994 and should be changed.
  - **LANMAXOUT** on the CTLD (for APPC environments): This parameter governs how often the sending system waits for an acknowledgement. Never allow LANACKFRQ on one system to have a greater value than LANMAXOUT on the other system. The parameter values of the sending system should match the values on the receiving system.

    Having appropriate values for the LANMAXOUT parameter along with the LAN acknowledgement frequency (LANACKFRQ) parameter for both the sending and receiving stations is essential to obtain optimal performance. Having other values may decrease throughput by up to 50% or even more if conditions trigger time-outs.

    In general, a value of *CALC (i.e., LANMAXOUT=2) offers the best performance for interactive environments, and adequate performance for large transfer environments.

    For large transfer environments, changing LANMAXOUT to 6 may provide a significant performance increase.
  - **LANWNWSTP for APPC** on the controller description (CTLD): If there is network congestion or overruns to certain target system adapters, then increasing the value from the default=*NONE to 2 or something larger may improve performance.
- When configuring an AS/400 system with communications lines and LANs it is important not to overload an IOP to avoid a possible system performance bottleneck.
- Optimally configured, the 100 Mbps Ethernet IOP/IOA can have an aggregate transfer rate of roughly 38 Mbps. Multiple concurrent large transfers may be required to drive the IOP at that rate (note that the test case provided had no associated data base activity or application processing).
- 100 Mbps Ethernet may provide a performance improvement over 10 Mbps Ethernet in terms of higher data rates for large transfer environments. It may also provide better overall performance within an establishment if the previous 10 Mbps Ethernet media was nearing its media utilization threshold.
- It is especially important to have the highest capacity IOP available for file serving, data base serving, or for environments that have many communications I/Os per transaction. This characteristic will also minimize the overall response time.
- Higher-performing TRLAN IOP/IOAs have the potential to overrun lesser capacity TRLAN IOP/IOAs. Many re-transmissions and time-out conditions

exist here. Check the AS/400 performance tools for these statistics. This can be minimized or avoided by limiting the LANACKFRQ and LANMAXOUT parameters to 1 and 2, respectively.

- For interactive environments it is recommended not to exceed 60% utilization on a LAN IOP. Exceeding this threshold in a large transfer environment or with a small number of concurrent users may still offer acceptable performance. Use the AS/400 performance tools to get utilization.
- A given model of the AS/400 system can attach multiple IOPs up to a given maximum number. It is important to distribute the workload across several IOPs if the performance capability of a single IOP is exceeded. There are also some limitations on the number of stations that can be configured through a single LAN connection.
- The larger mazimum frame size gives 16Mbit Token Ring emulation over ATM the advantage vs. Ethernet emulation over ATM. The round trip response times are equal but the large frame transfer rate is approximately 70% higher with token ring emulation.

5.3 APPC, TCP/IP, and Anynet

General APPC and TCP/IP Tips and Techniques

- In general TCP/IP and APPC provide similar performance. Some APPC APIs provide blocking (e.g., ICF and CPI-C), therefore scenarios that include repetitive small puts (that may be blocked) may achieve much better performance.
- For a given large transfer, using larger sends (record sizes) may provide a higher application data rate and decreased CPU time. With the larger record size, the CPU has less processing to do because of having fewer application reads and writes to transfer the same amount of data.
- The minimum round trip delay is about 3 milliseconds for TCP/IP or APPC using a LAN IOP. The CPW value for the CPU along with the load on the system will impact these times. These IOP delays are most noticeable in user transactions that contain many individual communications I/Os (like database serving). Having a fast IOP is critical to response time for these client/server environments.

  The round trip time increases as the size of the send/receive size increases. For example, a 32K send/receive may take 5 times longer than a 1000-byte send/receive (linear relationship).

  For this same scenario with SDLC, it takes about 50-100 ms for a 100-byte round trip. The longer transmission time on the media, the slower IOP, and the slower WAN media line turnaround (as compared to LAN), makes up the increased delay. Therefore, WAN and twinax connections usually yield poorer C/S performance.
- Application time for transfer environments, including accessing a data base file, decreases the maximum potential data rate. Because the CPU has additional work to process, a smaller percentage of the CPU is available to handle the transfer of data. For the CPB benchmark, doing data base accesses prior to each send cuts throughput by a factor of 3. The CPU time

requirements more than double because of the overhead of these data base accesses.

- Communications performance improvements that were introduced in V3R1 for APPC, TCP/IP, UDP/IP, FTP, and TELNET are discussed in *AS/400 Performance Capabilities Reference (V3R2), ZC41-8166*.

TCP/IP and FTP Tips and Techniques

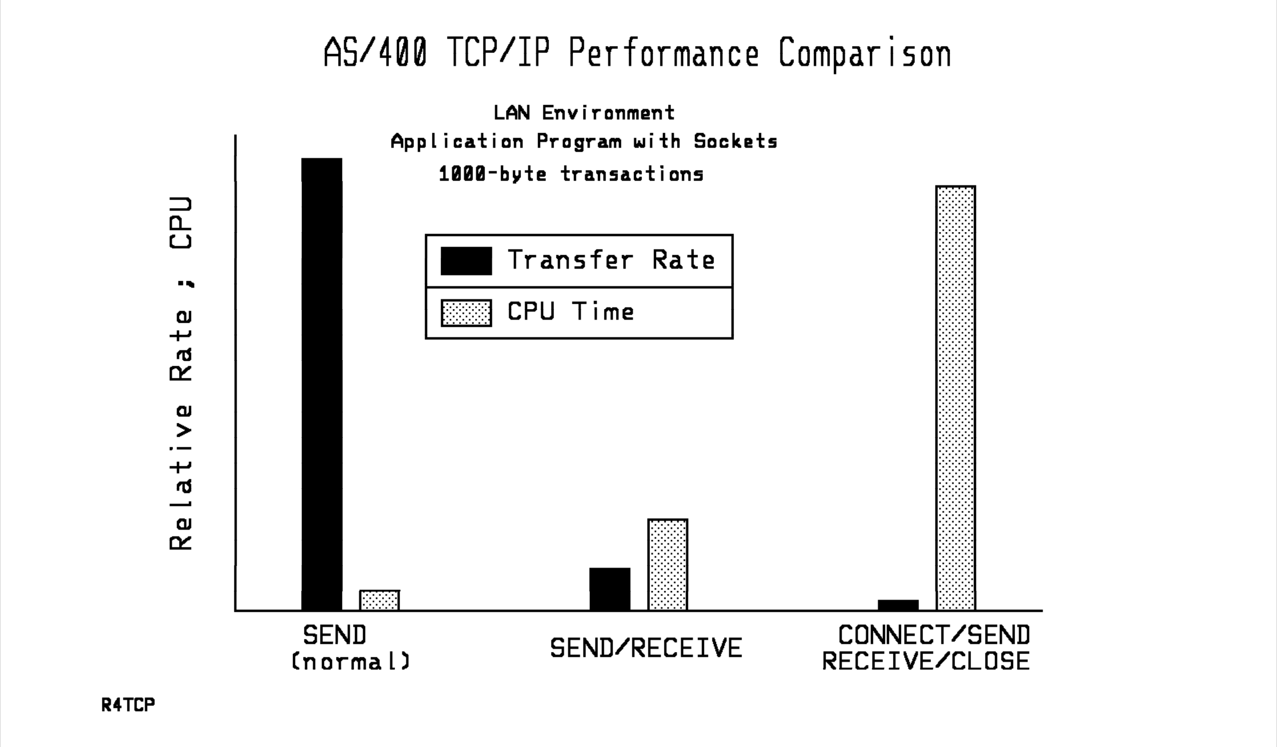


Figure 19. TCP/IP Performance

- From Figure 19, clearly the best way for an application to send data is to repetitively send large pieces of data without doing (intermitant) application acknowledgements or pacing. In other words, let the system-provided buffering and windowing functions deal with this.

- Connection and disconnections using sockets are significantly more expensive than normal sends and receives. Limit the number of times that new connections must be established. (i.e., leave the connection up if possible).

- TCP with large buffers (64K) out-performed UDP by 34% for large transfer rates. Although the UDP protocol is simpler than TCP, TCP uses a buffering mechanism which allows it to more effectively utilize the media.

- For optimum performance of UDP applications, the size of the send and receive buffers should be less than or equal to the MTU size of the media.

- In a send-only scenario, the TCP buffer size has a significant impact on performance. Larger buffer sizes up to 64K yielded substantially better performance. Buffer sizes larger than 64K provided a performance loss because too much time is spent filling the buffers serially.

- For optimum performance with FTP, make sure the MTU size is as large as possible. ASCII transfers are slower because of character conversion from

EBCDIC to ASCII prior to the data being sent. In a configuration similar to that of Table 19 on page 83, FTP transfers runs between 7 and 9 Mbps depending on the type of FTP transaction (GET vs PUT, file already exists, and replace vs new).

Sockets Performance Tips and Techniques

- Being sockets does not block up multiple applications sends, it is important to limit the number of interactions. Block in the application if possible.
- The **maximum frame size parameter** (MAXFRAME on LIND) can be increased from 1994 bytes for TRLAN (or other values for other protocols) to its maximum of 16393 to allow for larger transmissions.
- The **maximum transmission unit (MTU) size** parameter (CFGTCP command) for both the route and interface affect the actual size of the line flows. Increasing these values from 576 bytes to a larger size (up to 16388) will most likely reduce the overall number of transmissions, and therefore, increase the potential capacity of the CPU and the IOP.
- Increasing the **TCP/IP buffer size** (TCPRCVBUF and TCPSNDBUF on the CHGTCPA or CFGTCP command) from 8K bytes to 64K bytes may increase the performance when sending larger amounts of data.

APPC Tips and Techniques

- **MAXLENRU for APPC** on the mode description (MODD): If a value of *CALC is selected for the maximum SNA request/response unit (RU) the system will select an efficient size that is compatible with the frame size (on the LIND) that you choose. The newer LAN IOPs support IOP assist. Changing the RU size to a value other than *CALC may negate this performance feature.
- **Compression with APPC** should be used with caution and only for slower speed WAN environments. Many suggest that compression should be used with speeds 19.2 kbps and slower and is dependent on the data being transmitted (# of blanks, # and type of repetitions, etc.). Compression is very CPU-intensive. For the CPB benchmark, compression increases the CPU time by up to 9 times. RLE compression uses less CPU time than LZ9 compression (MODD parameters).
- Communications Error Recovery for APPC-Based PC Clients

  Communications error recovery for APPC-based PC clients improved about 40% in V4R1. The following enhancements contributed to the performance improvement:

  - Work Station Passthru job structure was redesigned to use ′server′ jobs as opposed to a communications job for every passthru session. This reduced the overhead in activation and deactivation by eliminating job initiation and termination.
  - File server communication was redesigned to use parallel jobs in the QSERVER subsystem as opposed to a single job for startup and takedown. This removed a system-wide bottleneck for Client Access users.
  - Target CNOS (Change Number of Sessions) was moved out of the system job QLUS (Logical Unit Services) to below the MI. So, the CPU time and I/Os for that portion of error recovery will now be charged to other jobs/tasks. This improved the APPC session start-up time by reducing MI crossings.

- Attach Manager was redesigned to allocate devices to a subsystem more quickly and use less storage per job. This improved environments where many APPC users connect at one time.

ICF and CPI-C (APPC) Tips and Techniques

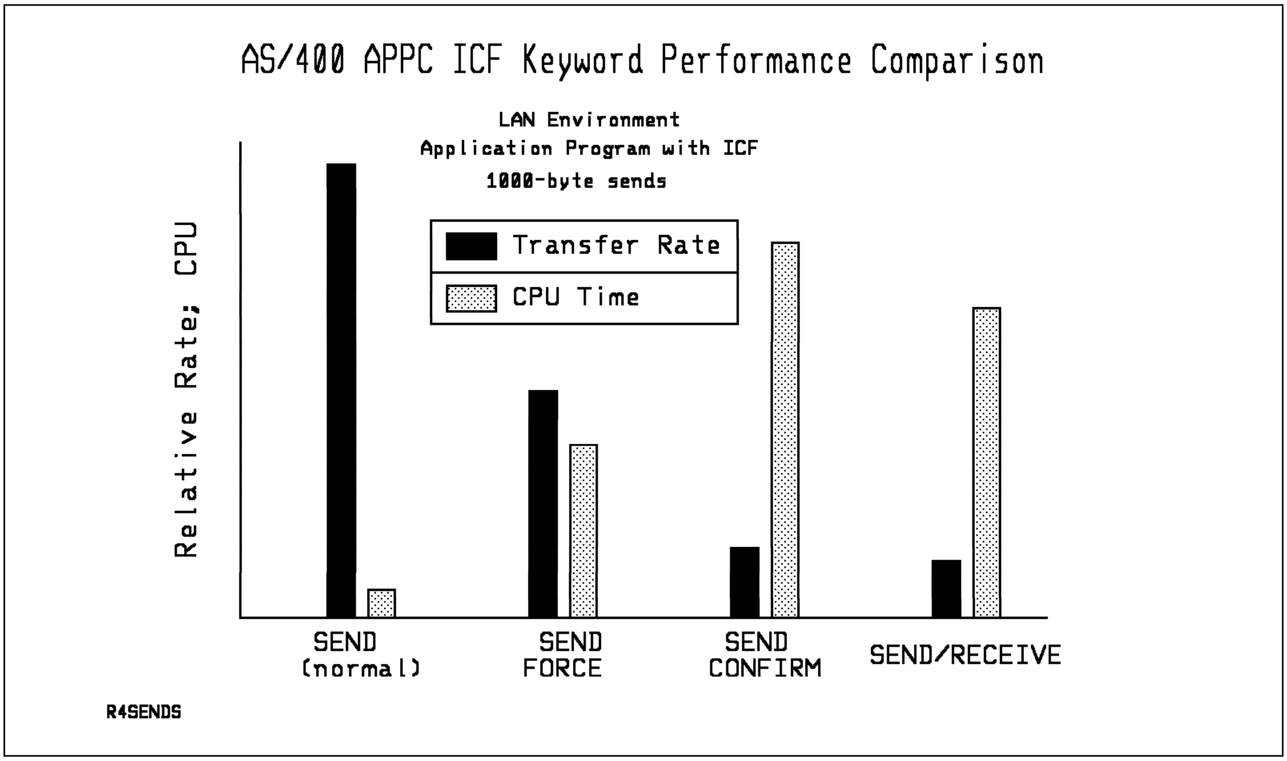


Figure 20. APPC/ICF Performance

- ICF and CPI-C have very similar performance for small data transfers.
- ICF allows for locate mode which means one less move of the data. This makes a significant difference when using larger records.
- The best case data rate is to use the normal blocking that OS/400 provides. For best performance, the use of the ICF keywords force data and confirm should be minimized. Note the performance comparison in Figure 20. An application′s use of these keywords has its place, but the trade-off with performance should be considered. Any deviation from using the normal blocking that OS/400 provides may cause additional trips through the communications software and hardware; therefore, it increases both the overall delay and the amount of resources consumed.
- When possible, reduce the number of API calls from the application.
  - Consolidate data before sending, e.g., combine an 8-byte status field with the 8K data, and send as one record.
  - Receive a complete record and parse it in your buffer. Don′t use multiple CMRCV′s to receive a single record
  - Don′t do a 0 byte receive to get the status to tell you if you have to do a receive.
  - Use Confirm only when necessary
  - Use Request To Send only when necessary

- Use Invite only when soliciting input from *multiple* devices, otherwise use Read instead
- If using Invite to solicit from multiple program devices, follow it with a Read-from-invited operation, not a Read operation.

- For best ICF performance:
  - Eliminate unused record formats.
  - Use separate record formats instead of multi-purpose record formats with option indicators.
  - Code to use the same record format for repeated operations.
  - Set maximum program devices = 1
  - use non-shared file
  - use separate indicator area

Anynet Tips and Techniques

- Having ANYNET = *YES causes extra CPU processing. Only have it set to *YES if it is needed functionally; otherwise, leave it set to *NO.
- For send and receive pairs, the most efficient use of an interface is with it′s ″native″ protocol stack. That is, ICF and CPI-C perform the best with APPC, and Sockets performs best with TCP/IP. There is CPU time overhead when the ″cross over″ is processed.
- Each interface/stack may perform differently depending on the scenario. The bottom of the APPC stack is slightly more efficient than the TCP stack. ICF and CPI-C APIs provide buffering which reduces overall CPU time; sockets does not provide this level of buffering.

DDM, ODF, and FTS Tips and Techniques

- Copyfile with DDM provides an efficient way to transfer files between AS/400s. DDM provides large blocking which limits the number of times the communications support is invoked. It also maximizes efficiencies with the data base by doing fewer larger I/Os. Generally, a higher data rate can be achieved with DDM compared with user-written APPC programs (doing data base accesses) or with ODF.
- When ODF is used with the SNDNETF command, it must first copy the data to the distribution queue on the sending system. This activity is highly CPU-intensive and takes a considerable amount of time. This time is dependent on the number and size of the records in the file. Sending an object to more than one target AS/400 only requires one copy to the distribution queue. Therefore, the realized data rate may appear higher for the subsequent transfers.
- FTS is a less efficient way to transfer data. However, it offers built in data compression for linespeeds less than a given threshold. In some configurations, it will compress data when using LAN; this significantly slows down LAN transfers.

5.4 Work Station Attachment Comparison

There are many ways to attach work stations (WS) to the AS/400 via communications. Each type can have different overheads and unique performance characteristics.

Interactive transactions include CPU processing for WS connection and application processing. If the application is ″light″ (not very much processing in a relative sense, similar to RAMP-C), then the performance impact of how WS′s are connected can be greater. If the application is ″heavier″ (significantly more CPU processing) the performance impact of connectivity type is not as significant.

The Following graph illustrates the relative performance of several types of WS attachment with the CPU resource being the limiting factor:

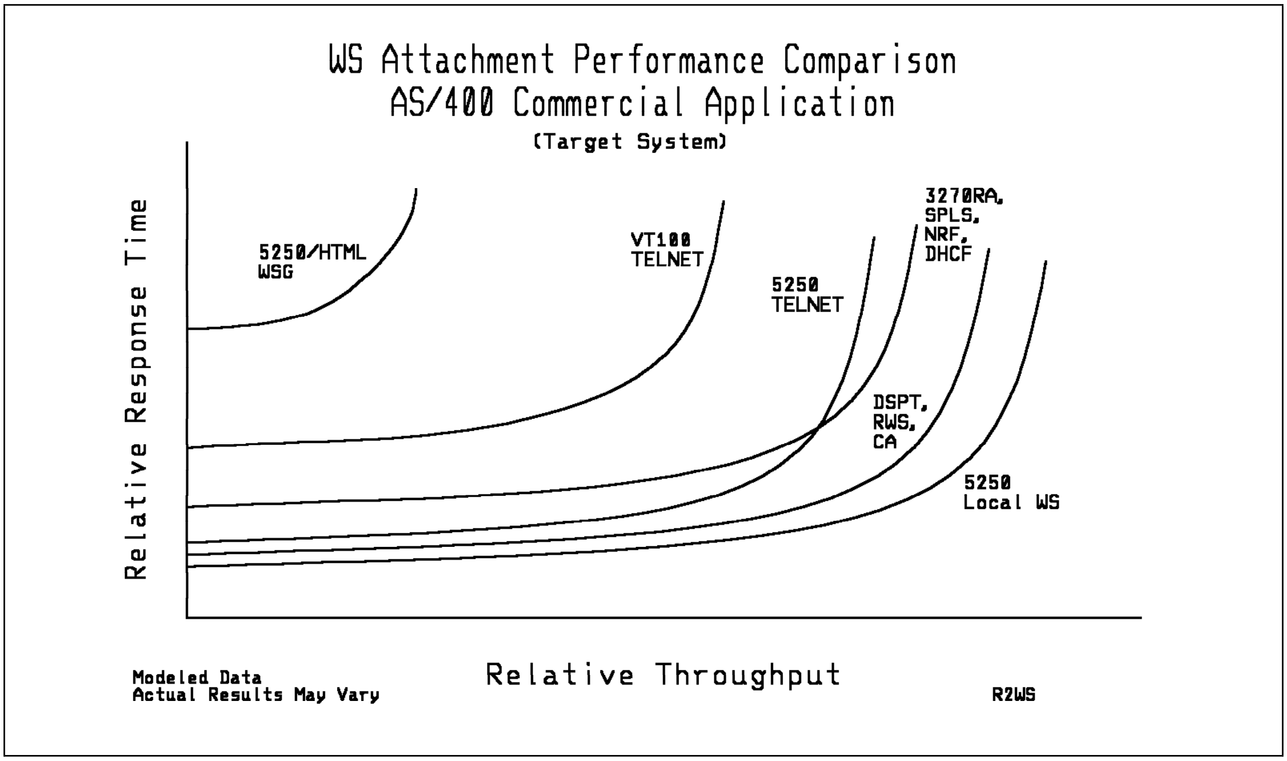


Figure 21. Work Station Attachment Comparison

Work Station Tips and Techniques

- **Attaching WS through communications has more CPU overhead than 5250 local WS.** Keep in mind that the actual overhead may vary significantly with changes in the data stream (number of I/Os, number of bytes, number of fields, and other screen I/O characteristics). For example, compared with local 5250 WS:
  - 5250 DSPT (target side) impacts overall AS/400 capacity by 5-10% due to additional CPU processing for communications per transaction.
  - Remote work stations with the 5494 controller impacts overall AS/400 capacity by 5-10% due to additional CPU processing for communications

per transaction. This is for either 5250 WS attached remotely or for PWS-type WS attached to the 5494 (either remotely or via LAN).

  - WS attached via CA/400 impacts capacity by about 10% due to additional CPU processing for communications per transaction.
  - TELNET impacts overall AS/400 capacity by 20% or more due to additional CPU processing for TCP/IP communications per transaction. Note that using the IBM Network Station as a work station uses this method of attachment.
  - VT100/VT220 reduces overall AS/400 capacity by about 40% due to additional CPU processing for TCP/IP, data stream translation. This impact can vary greatly due the characteristics of the data stream (# bytes, # fields, function keys, etc.).
  - 3270 Remote Attach, DHCF, NRF, and SPLS impact overall AS/400 capacity by 15-20% due to additional CPU processing for communications and data stream translation per transaction. This impact can vary greatly due the characteristics of the data stream (# bytes, # fields, complexity, etc.).
  - 5250/HTML Workstation Gateway reduces overall AS/400 capacity by about 4 times due to additional CPU processing for communications, the WSG servers, and 5250 to HTML conversions.

- **Passing Through an AS/400 to an application on another system** has several possibilities:
  - 5250 DSPT (source side) is the baseline here. This ″front-end″ system only has to support the WS attachment processing and communications support to the server system with no application processing being it is just a source side.
  - TELNET (source side) uses 3 times more CPU time than 5250 DSPT because of additional processing for TCP/IP communications and more processing in the TELNET application.
  - APPN intermediate node routing (intermediate system passing transaction from source to target) has a similar CPU time to 5250 DSPT (source).
- **Twinaxial controllers provide better performance than ASCII controllers.** This is primarily due to the increased line speed. The conversion from ASCII to EBCDIC is performed in the ASCII controller, so it is not an impact to AS/400 CPU time. ASCII response time should be similar to the response time for a remote work station configuration with a similar line speed.
- Keep the line utilization below 30% for best performance when interactive users are attached. This will maintain predictable and consistent response times. Exceeding 50-60% line utilization will usually cause unacceptable response times.
- **Mixed interactive users and batch:** When interactive users and large transfers are running on a communications line concurrently, consider the following to keep interactive performance acceptable:
  1. Use APPN transmission priority to prioritize the interactive users′ transfers over that of the large transfer.
  2. Change the RU size to a lower value for the large transfer. This optimizes interactive response time at the expense of large transfer

performance (note that overall CPU time will increase also for the large transfer).

3. Reducing the pacing values for the large transfer will also slow it down, allowing the interactive users more windows for getting on the line.
4. Note that the 2nd and 3rd items here also penalize the large transfer if there is no interactive traffic. Therefore, the 1st item is preferred.

5.5 Opti-Connect for OS/400

OptiConnect/400 (OC/400) enables applications to transparently access databases on remote AS/400 systems. It allows applications written to access databases locally, to access them remotely with simple changes to file descriptions and no changes to the applications.

OC/400 provides an optimized DDM solution capable of efficient, low-latency, high bandwidth communication between AS/400 systems. OC/400 utilizes a fiber optic connected shared I/O bus and a SLIC shared bus device driver to transport data and messages. The efficiency of OC/400 allows horizontal growth solutions - increasing total capacity to shared data beyond the bounds of a single system. Prior to OC/400, DDM using ICF/APPC for transport has provided the remote data access function but the overhead of communication between systems was too high to achieve horizontal growth under heavy workloads. This overhead kept DDM from being a viable solution for user′s horizontal growth. OC/400 offers customers this horizontal growth by enabling 2 to 32 AS/400 systems to share an I/O bus. Any or all systems can act as both an application machine (workstation server) for accessing another systems data or a database server providing data to other systems.

The following table shows system capacity growth potential when OC/400 is used in system environments with various levels of data base activity. This table only applies to the work that is being considered for OC/400 usage. The database server machine is assumed to have similar configuration like the single system has. These are only guidelines and results will vary depending on system environments. These guidelines were based on data from a previous release, but it is assumed that the relative ratios would still hold. The growth factors will also vary with the workload (read/write ratios, open/close frequency, etc.).

Table 20. OC/400 Horizontal Growth Potential

OC/400 Horizontal Growth Potential
Single Database Server and Multiple Application Machines (1 to 7)

DB %	1-A	2-A	3-A	4-A	5-A	6-A	7-A
10	1.81	2.63	3.44	4.26	5.07	5.89	6.70
20	1.66	2.31	2.97	3.62	4.17	-	-
30	1.52	2.03	2.55	2.78	-	-	-
40	1.39	1.79	2.08	-	-	-	-
50	1.29	1.57	1.67	-	-	-	-
60	1.19	1.38	1.39	-	-	-	-

Note:

- DB% = The percent CPU time allocated to data base activity in a single system environment.
- 2-A = 3 systems composed of 2 application machines and 1 database server
- Client Factor = 1.8 for this workload (ie, the OC/400 CPU associated with the client. eg, offloading 10% database would be replaced by 18% OC/400 CPU).
- Server Factor = 1.2 for this workload (ie, the OC/400 and related database CPU associated with the server. eg, offloading 10% database from the client would represent 12% CPU on the server).
- A typical customer environment includes 1 database server and and 2 or more application machines.
- The columns show the capacity gain over a single system. For example, 20% DB with 2-A shows a value of 2.31. This value is the capacity growth achieved by using 3 systems; 1 as server and 2 as application machines over a single system. Any remaining capacity on the server not directed at the client(s) can do additional single system workload.
- All systems are assumed to have the same CPW value for this exercise.

Tips and Techniques

1. Each time that a ′put′ or ′get′ is issued, OC/400 uses the shared I/O bus to exchange data with the remote system. DDM uses a communication link for accessing data on remote system.

2. V4R1 OC/400 performance should be equal or better than V3R6 (and also V3R1) when the capacity is scaled to CPW of the CPU.

3. OC/400 is a more efficient way than DDM (with communications) to access data on another (near by) AS/400.

4. At higher levels of throughput the system overhead for OC/400 increases in a non-linear fashion. This is due to multiprocessing contention when CPU utilization is higher. It is recommended that CPU utilization should not exceed 90% on either the application machine or database server system for this reason.

5. There is an overhead (called client/server factor) associated with using OC/400 both on the application and data base machine. This overhead varies and is dependent on factors like:

    - Percent of data base activity
    - Number of logical IOs per transaction
    - Usage of Journal and Commitment Control

- Number of data base Open/Close

6. The Maximum length for the OC/400 fiber optic cables to the shared bus is 2KM, requiring all OC/400 systems to be within 2KM of the system providing the shared bus. Due to fiber latency of 5 nano-seconds per meter, it is recommended that systems should be within 300 meters of each other. This may avoid any performance impact due to line latency. DDM using APPC over LANs or WANs can provide greater distances.

7. The OC/400 adapter′s capacity may be exceeded with the new V4R1 CPU models. You may use WRKOPCACT command or the PFRMON (logical I/O field) to observe the OC/400 ops/sec rate. From measurements with a workload that did small read operations (data not provided here), the following maximums were observed:

   - Features 2682/2685/2688 (20M cable): 10,200 ops/sec (1.8 Mbps)
   - Features 2682/2685/2688 (500M cable): 7,100 ops/sec (1.2 Mbps)
   - Features 2680/2683/2686 (2KM cable): 2,700 ops/sec (0.5 Mbps)

   On an 8-way model 650 with the workload mentioned above, two OC/400 adapters were needed to fully drive the 8-way to full capacity. The PFRMON does not show the utilization of this adapter in the reports. So you should consider these maximums to see if the adapter is possibly the bottleneck. If so, adding additional adapters and cabling is possible.

   The 2680/2683/2686 features provide a higher level of performance because of being associated with cables of 500M or less.

   The length of the cable also impacts OC/400 capacity. Note from above that throughput drops with cable length by up to 30% going from the short to long cable with the 2680/2683/2686 features.

8. The OC/400 horizontal growth is achieved across multiple systems with some or all systems acting as servers. Any single database network of logical and physical files must reside on a single system, therefore multiple servers require multiple, separate database networks. The maximum load supported across multiple systems is typically limited by the capacity of a database server - additional application systems can be added until the database server saturates.

9. When analyzing batch workloads, more attention must be paid to logical database I/O′s, blocking, and pool size. Performance characteristics may be different for batch versus interactive use of OC/400. In particular, batch jobs are often very database intensive and accumulated CPU and delays for each logical I/O which must traverse the OC/400 shared bus can add significant CPU utilization and elapsed time to batch runs.

10. A detailed analysis to determine where to place the data files is very critical in batch workloads or read only files.

11. An OC/400 shared bus adapter can support up to 40 MBytes/second data rate. Measured bus/adapter utilization in majority of OC/400 testing has shown CPU or adapter limits are reached prior to reaching bus/adapter limits

12. In most distributed environments, including OC/400, there is a penalty for remote access; obviously this penalty is most severe when 100% of accesses are remote and may become negligible for lower percentages (eg. 10%). Therefore an optimum environment should minimize remote access by distributing users or data across systems in a manner to maximize local access. Replication of read-only files to application machines, distribution of

data across multiple database servers, and location of batch work on application machines or database servers must be carefully considered when defining an OC/400 environment to optimize total system capacity.

Chapter 6. Web Serving Performance

Performance information for web serving on the AS/400 and various types of web server transactions will be discussed in this section.

With V4R3, the IBM HTTP Server is used as the web server. Prior to V4R3, the Internet Connection for the AS/400 (IC/400) was used. Web serving with the IBM HTTP Server on V4R3 performs up to 60% better than V4R2 (holding hardware constant).

There are many factors that can impact overall performance (e.g., end-user response time, throughput) in the complex Internet environment, some of which are listed below:

- Web Browser
  - processing speed of the client system
  - performance characteristics of the Web browser
  - client application performance characteristics
- Communications network
  - speed of the communications links
  - capacity of any proxy servers
  - congestion of network resources
- AS/400 Web server
  - AS/400 processor speed
  - utilization of key AS/400 resources (CPU, IOP, memory, disk)
  - Web server performance characteristics
  - application (e.g., CGI) performance characteristics

The primary focus of this section will be to discuss the performance characteristics of the AS/400 as a server in a Web serving environment, providing capacity planning information and recommendations for best performance. Please refer to Chapter 5, “Communications Performance” on page 79, for related information.

Data accesses across the Internet differ distinctly from accesses across ′traditional′ communications networks. The additional resources to support Internet transactions by the CPU, IOP, and line are significant and must be considered in capacity planning. Typically, in a traditional network:

- there is a request and response (between client and server)
- connections/sessions are maintained between transactions
- networks are tuned to use large frames

For web transactions, there are a dozen or more line transmissions (including acknowledgements) per transaction:

- a connection is established for each transaction
- there is a request and response (between client and server)
- the connection is closed after each transaction
- networks typically have small frame (MTU) sizes
- HTML screens typically generate more bytes than traditional
- one user transaction may contain separate internet transactions
- secure transactions are frequent and consume more resource

The information that follows is based on performance measurements and analysis done in the internal IBM performance lab. The raw data is not provided here, but the highlights, general conclusions, and recommendations are included. Results listed here do not represent any particular customer environment. Actual performance may vary significantly from what is provided here. Note that these benchmarks, along with other published benchmark data (from other sources) are always measured in best-case environments (e.g., local LAN, large MTU sizes). Real internet networks typically have higher contention, MTU size limitations, and intermediate network servers (e.g., proxy, SOCKS).

6.1 Web Serving with the HTTP Server

The Hypertext Transfer Protocol (HTTP server) allows AS/400 systems attached to a TCP/IP network to provide objects to any Web browser. At a high level, the connection is made, the request is received and processed, the data is sent to the browser, and the connection is ended. The HTTP server jobs and the communications router tasks are the primary jobs/tasks involved (there is not a separate user job for each attached user).

Workload Description and Data Interpretation: The workload is a program that runs on a client work station. The program simulates multiple Web browser clients and repetitively issues ′URL requests′ to the AS/400 Web server. The number of simulated clients can be adjusted to vary the offered load. Each of the transaction types listed in the tables serve about 1000 bytes:

- SPS: serves a static page via the HTTP server. This information can be accessed from the web server′s cache of specified IFS files.
- CGI (HTML): invokes a CGI program that accesses data from IFS and serves a simple HTML page via the HTTP server. This runs in a named activation group.
- CGI (SQL): invokes a CGI program that performs a simple SQL request and serves the result via the HTTP server. This runs in a named activation group.
- Persistent CGI: invokes a CGI program that receives a handle supplied by the browser, accesses data from IFS, and serves a simple HTML page via the HTTP server.
- Net.Data (HTML): invokes the Net.Data program that serves a simple HTML page via the HTTP server.
- Net.Data (SQL): invokes the Net.Data program that performs a simple SQL request and serves the result via the HTTP server.
- Net.Commerce: invokes the Net.Commerce service program to do a simple browse transaction and then serves the result via the HTTP server. This can be accessed from the Net.Commerce cache if configured appropriately.

Each of the above can be served in secure or non-secure fashion. *″CPU time per transaction″* is the average AS/400 CPU time to process the user transaction for the specific scenario. *″AS/400 Capacity (hits/sec/CPW)″* is the capacity metric used to estimate the capacity of any AS/400 model. An example exists in the conclusions. *″IOP Capacity″* represents the maximum transaction rate that a single IOP could handle.

The CGI programs were compiled using a ″named″ activation group. For more information on program activation groups refer to *AS/400 ILE Concepts, SC41-5606*.

Web Serving Performance Measurements: The following tables provide a summary of the measured performance data. These charts should be used in conjunction with the rest of the information in this section for correct interpretation. Results listed here do not represent any particular customer environment. Actual performance may vary significantly from what is provided here.

Table 21. V4R3 AS/400 Web Serving

Transaction Type:	Non-secure		Secure	
	Capacity Metric: hits/sec/cpw	Relative CPU Time	Capacity Metric: hits/sec/cpw	Relative CPU Time
Static Page (cached)	1.25	0.8 x	0.54	1.9 x
Static Page (not cached)	1.00	1.0 x	0.44	2.2 x
CGI (HTML)	0.26	3.8 x	0.20	5.0 x
CGI (SQL)	0.26	3.8 x	0.20	5.0 x
Persistent CGI	0.27	3.7 x	0.21	4.7 x
Net.Data (HTML)	0.14	7.1 x	0.12	8.3 x
Net.Data (SQL)	0.11	9.0 x	0.10	10.0 x
Net.Commerce (cached)	0.25	4.0 x	n/a	n/a
Net.Commerce (not cached)	0.006	167 x	n/a	n/a

Note:

- IBM HTTP Server for AS/400; V4R3; 16Mbps TRLAN
- Based on measurements from an AS/400 Model 170-2386
- Net.Commerce is estimated based on V4R2 code
- Static page caching done with IBM HTTP server (WRKHTTPCFG)
- Net.Commerce caching done with Net.Commerce internal cache
- 1KB data is served for each of the transaction types
- data assumes no access logging
- Static page serving is served from the root directory of IFS
- CGI programs compiled with ″named″ activation group
- Secure measurements done using Secure Sockets Layer (SSL) with 40-bit RC4 encryption
- CPW is the ″Relative System Performance Metric″ found in Chapter 2, “AS/400 System Capacities and CPW” on page 3.
- Web server capacities may not scale exactly by CPW
- results may differ significantly from those listed here.
- n/a = not available

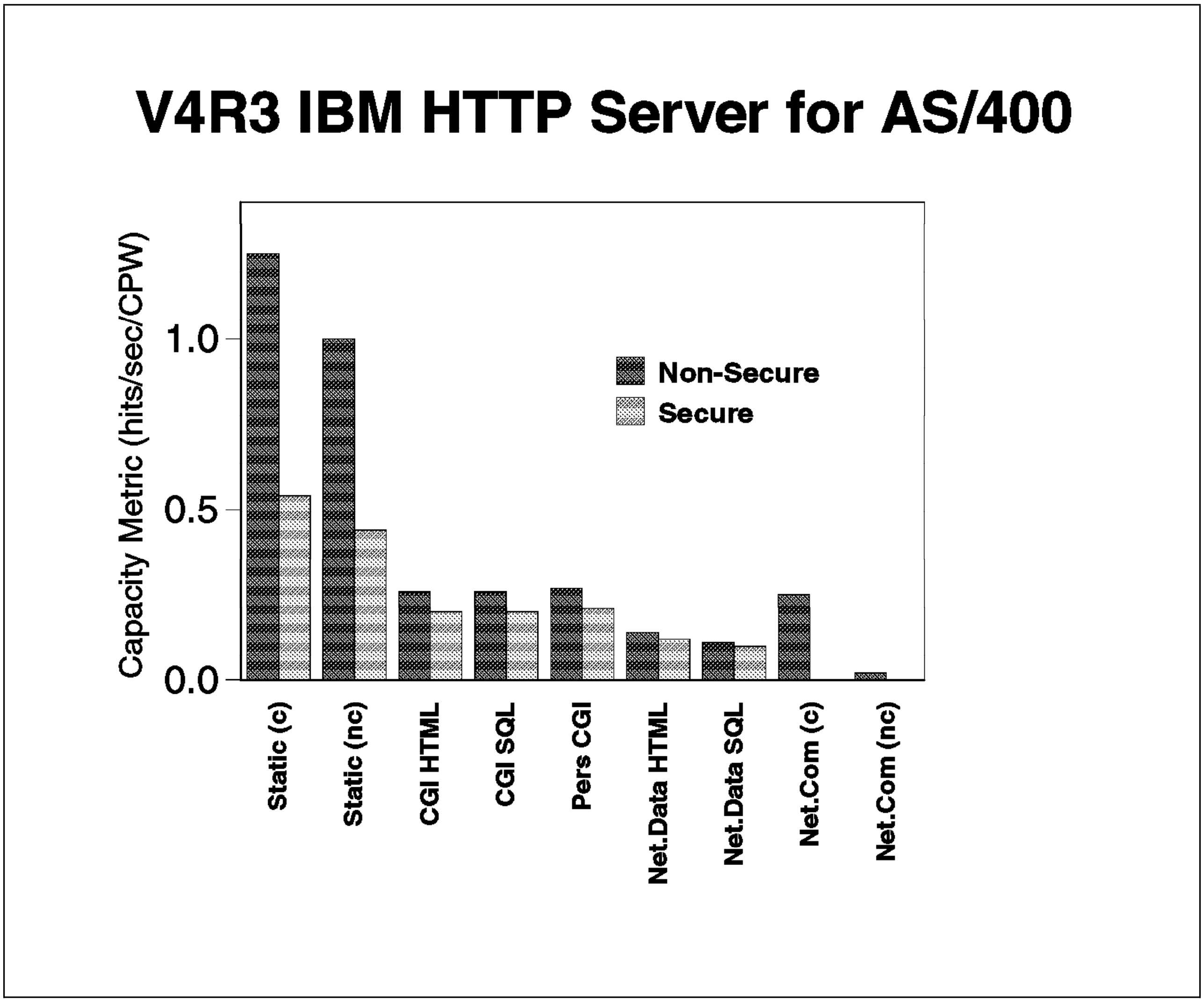


Figure 22. AS/400 Web Serving V4R3 Relative Capacities

Web Serving Performance Tips and Techniques:

1. **V4R3** provides a performance improvement in capacity of up to 65% over that of V4R2 (with similar hardware). This is mostly due to the improved efficiency of the IBM HTTP Server over that of the ICS/400 from V4R2. For static pages that are not cached, V4R3 provides up to 20% more capacity. For static pages that are cached, V4R3 provides up to 65% more capacity. There are also significant performance improvements for Net.data and CGI with named activations in V4R3.

   V4R2 provided similar performance to V4R1.

   V4R1 provided a performance improvement in capacity of up to 50% over that of V3R7 (with similar hardware). This is primarily due to improvements in TCP/IP performance.

2. **Web Serving Capacity (Example Calculation):** Throughput for web serving is typically discussed in terms of the number of hits/second (connections/second and transactions/second also mean the same). Typically, the CPU will be the resource that limits capacity. If a larger AS/400 model is used with a single IOP, then the IOP may be the limiting factor.

For system configurations where the CPU is the limiting resource, the tables above can be used to estimate capacity for a given system. Simply multiply the CPW (relative system performance metric Chapter 2, “AS/400 System Capacities and CPW” on page 3) for the AS/400 model with the appropriate hits/second/CPW value (provided in the tables above). For example, a S20-2161 rated at 113.8 CPW doing web serving with CGI programs, would have a capacity of 30 hits/sec (113.8 x 0.27 = 30). This assumes that the entire capacity of the system would be allocated to Web serving. If other work will also be on the system, you must pro-rate the CPU allocation. For example, if only 25% of the CPU is allocated for Web serving, then it would have a web serving throughput of 7 hits/sec (113.8 x 0.27 x 25% = 7).

Use these high-level estimates with caution. They do not take the place of a complete capacity planning session with actual measurements of your particular environment. Scaling issues for the server, the application, and the database also may come into consideration when using N-way processors with higher projected capacities.

3. **Web Server Cache for IFS Files:** This is new for V4R3. Serving static pages that are cached can increase web server capacity by almost 40%. Ensure that highly used files are selected to be in the cache (WRKHTTPCFG).

4. **Page size:** The data in the tables assumes about 1K bytes being served. If the pages are larger, more bytes are processed, CPU processing per transaction increases,and therefore the transaction capacity metrics would be reduced. For larger pages, a rough estimate would be to assume 3 times the CPU resource listed in the table for every 200K-byte increase in page size. Likewise, if the transaction is secure (encryption costs), assume that the overhead is 6 times the CPU resource listed in the table for every 200K-byte increase in secure page size.

5. **CISC vs. RISC** The HTTP serving environment on RISC AS/400 models provides better performance (per CPW) than on CISC AS/400 models. The hits/sec/CPW factor for RISC models (listed above) is almost twice that of the CISC models (listed in the V3R2 version of this document). This is due to the better RISC technology for C-language environments and to recent performance improvements to TCP/IP and the web server.

   Note that CPWs (relative system performance metrics) are derived from a commercial workload′s capacity. Therefore, relative capacity ratios for Web serving may not track accordingly. However, they should be adequate for a high-level capacity planning exercise.

6. **Response Time (general):** User response time is made up of Web browser (client work station) time, network time, and server time. A problem in any one of these areas may cause a significant performance problem for an end-user. To an end-user, it may seem apparent that any performance problem would be attributable to the server, even though the problem may lie elsewhere.

   It is common for pages that are being served to have imbedded images (e.g., gifs). Each of these separate Internet transactions adds to the response time since they are treated as independent HTTP requests and can be retrieved from various servers (some browsers can retrieve multiple URLs concurrently).

7. **HTTP and TCP/IP Configuration Tips:**

   a. The **number of HTTP server jobs**: The CHGHTTPA command has parameters that specify the minimum and maximum number of server jobs. This is a system-wide value. The WRKHTTPCFG command also can specify similar values (MaxActiveThreads and MinActiveThreads). These values would override the values that are set via CHGHTTPA and would be for a given configuration. The reason for having multiple server jobs is that when one server is waiting for a disk or communications I/O to complete, a different server job can process another user′s request. Also, for N-way systems, each CPU may simultaneously process server jobs. The system will adjust the number of servers that are needed automatically (within the bounds of the minimum and maximum parameters).

The values specified are the number of ″child or worker″ threads. Typically, 5 server threads are adequate for smaller systems (100 CPWs or less). For larger systems dedicated to HTTP serving, increasing the number of servers to 10 or more may provide better performance. A starting point for the maximum number of threads can be the CPW value divided by 20. Try not to have more than what is needed as this may cause unnecessary system activity.

b. The **maximum frame size parameter** (MAXFRAME on LIND) can be increased from 1994 bytes for TRLAN (or other values for other protocols) to its maximum of 16393 to allow for larger transmissions. Typically documents are larger than 1994 bytes.

c. The **maximum transmission unit (MTU) size** parameter (CFGTCP command) for both the route and interface affect the actual size of the line flows. Increasing these values from 576 bytes to a larger size (up to 16388) will most likely reduce the overall number of transmissions, and therefore, increase the potential capacity of the CPU and the IOP.

   Similar parameters also exist on the Web browser. The negotiated value will be the minimum of the server and browser (and perhaps any bridges/routers), so increase them all.

d. Increasing the **TCP/IP buffer size** (TCPRCVBUF and TCPSNDBUF on the CHGTCPA or CFGTCP command) from 8K bytes to 64K bytes may increase the performance when sending larger amounts of data. If data coming into the server is simply requests, increasing TCPRCVBUF may not provide any benefit.

e. **Secure Web Serving:** Secure web serving involves additional overhead to the server. Additional line flows occur (fixed overhead) and the data is encrypted (variable overhead proportional to the number of bytes). Note the capacity factors in the tables above comparing non-secure and secure serving. For simple transactions (e.g., static page serving) the impact of secure serving is 2x or more based on the number of bytes served. For complex transactions (i.e., CGI or Net.Data), the overhead in the range of 20%.

f. E-Business applications typically yield a variety of complex transactions. These transactions have sub-transactions made up of static pages, CGI, Net.Data, etc. Capacity planning for these is more complex and warrants a careful analysis of the make-up of the transactions. The data from tables can assist with this analysis.

g. **Error and Access Logging:** Having logging turned on causes a small amount of system overhead (CPU time, extra I/O). Turn logging off for best capacity. Use the WRKHTTPCFG command to make these changes.

h. **Name Server Accesses:** For each Internet transaction, the server accesses the name server for information (IP address and name translations). These accesses cause significant overhead (CPU time, comm I/O) and greatly reduce system capacity. These accesses can be be eliminated by using the WRKHTTPCFG command and adding the line ″DNSLookUp Off″.

8. **HTTP Server Memory Requirements:** Follow the faulting threshold guidelines suggested in the work management guide by observing/adjusting the memory in both the machine pool and the pool that the HTTP servers run in (WRKSYSSTS).

   Factors that may significantly affect the memory requirements include using larger document sizes, using CGI.bin programs, and using Net.Data.

9. **AS/400 Model Selection:** Use the information provided in this section along with the characterization of your HTTP workload environment in a capacity planning exercise (perhaps with BEST/1) to choose the appropriate AS/400 model. All the tasks, jobs, and threads associated with HTTP serving are ′non-interactive′, so AS/400e servers or AS/400 Advanced Servers would provide the best price/performance (unless other interactive work is present on the system).

10. **File System Considerations:** Web serving performance varies significantly based on which file system is used. Each file system has different overheads and performance characteristics. Note that serving from the ROOT or QOPENSYS directories provide the best system capacity. If Web page development is done from another directory, consider copying the data to a higher-performing file system for production use.

    The web serving performance of the non-thread-safe file systems is significantly less than the root directory. Using QDLS or QSYS may decrease capacity by 2-5 times. For a more detailed discussion of IFS performance, please refer to the IFS section from the V3R2 version of this document.

11. **File Size Considerations:** The connect and disconnect costs are similar regardless of size, but cost for the transmission of the data with TCP/IP and the IFS access vary with size. As file size increases, the IOP is more efficient by being able to achieve a higher aggregate data rate. However, being larger, the files require more data frames, thus causing the hits/sec capacity for the IOP to go down accordingly.

12. **Communications/LAN IOPs:** Since there are a dozen or more line flows per transaction, the Web serving environment utilizes the IOP more than other communications environments. Use the performance monitor (STRPFRMON) and the component report (PRTCPTRPT) to measure IOP utilization. Attempt to keep the average IOP utilization at 60% or less for best performance.

    IOP capacity depends on file size and MTU size (make sure you increase the maximum MTU size parameter). Additional information on communications/LAN IOP performance can be found in section 5.2, “LAN Protocols, Lines, and IOPs” on page 82.

    The 2619 or the 2617 LAN IOPs have a capacity of roughly 70 hits/sec when serving small (e.g., 1K byte) nonsecure pages (keep in mind that each hit contains a dozen or so line flows). Using Ethernet or TRLAN IOPs from V4R1 or more recent, have capacities in the 100-130 hits/sec range.

On larger AS/400 models, the comm/LAN IOP may become the bottleneck before the CPU does. If additional HTTP capacity is needed, multiple IOPs (with unique IP addresses) could be configured. The overall workload would have to be ′manually′ balanced by Web browsers requesting documents from a set of interfaces. The load can also be balanced across multiple IP addresses by using DNS (domain name server).

13. **Net.Data and DB2WWW:**

    - Since Net.Data is the follow-on to DB2WWW version 1, ensure that you have the right PTFs applied to get the new version. See http://www.as400.ibm.com/netdata for the latest code levels and more product information.
    - Net.Data performs significantly better than DB2WWW version 1. Both response time and capacity improved by more than a factor of two.
    - Net.Data is more disk I/O intensive than typical HTTP transactions. Therefore, more HTTP server jobs may be needed to provide the optimal level of system throughput.
    - A Net.Data SQL macro is slightly slower than an SQL CGI.bin. This is because the SQL macro does some formatting while the SQL CGI.bin does no formatting. There are functional advantages in using an SQL macro.
        - direct reuse of existing SQL statements (no programming required)
        - provides the built-in ability to format SQL results
        - provides the ability to store SQL results in a table and pass the results to a different language environment (e.g., REXX).
    - If macro performance is critical, turn off the reporting function (uses more CPU time).
    - For REXX macros, calling an external REXX program will perform better than inline REXX statements.

14. **CGI and Persistent CGI:** Significant (perhaps as much as 2x) performance benefits can be realized by compiling into a ″named″ versus a ″new″ activation group. Refer to the *AS/400 ILE Concepts* for more details on activation groups. Persistent CGI is specific to applications needing to keep state information across web transactions. Don′t confuse persistent CGI with a way to improve the performance of your CGI program. You′ll notice in the earlier table that the performance of CGI is nearly identical to that of persistent CGI.

6.2 5250/HTML Workstation Gateway (WSG)

5250/HTML Workstation Gateway enables all Web browsers to be clients to existing 5250 applications without making changes to the application. At a high level, the connection is made, the request is received and processed, the user job processes the application, the workstation I/O is converted from 5250 to HTML, the HTML is sent to the browser, and the connection is ended. The tasks involved are the WSG server jobs, the user job (one per client), the communications router tasks, the virtual terminal task, and the Telnet task.

WSG Performance Tips and Techniques:

1. **WSG CPU Time and Capacity:** WSG involves a significant amount of CPU processing compared with other workstation connectivity types (e.g., Telnet).

Typical commercial transactions with WSG use 3 times the CPU time of Telnet running the same application (on the target system). Therefore, it is recommended to use WSG casually or for non-performance critical transactions. It would not be a good price/performance decision to convert all the workstations in an existing production environment from twinax or communications to WSG. (Note that WSG uses 6 times the CPU time of Telnet in CISC models, but only 3 times more CPU time in RISC models.)

2. **WSG Response Time:** Response time is made up of browser time, network time, and server time. Browsers typically contribute 0.5 - 2 seconds to response time, network time can vary widely, and AS/400 server time typically ranges from 0.2 - 5 seconds depending on the model, load, and application.

   A problem in any one of these areas can cause a significant performance problem for an end-user. To an end-user, it may seem apparent that any performance problem would be attributable to the server, even though the problem may lie elsewhere.

   Better response time may be achieved by using the STYLE button on the browser (WSG window) to remove the F-keys on the bottom. This reduces the the browser time to process and display the screen.

3. **WSG Configuration:** Default settings for the WSG related parameters typically will work fine. Changing some of the following parameters may improve performance.

   - The number of clients per WSG server job (CHGWSGA command): This parameter controls the maximum number of users that can be attached to a given WSG server job. When the WSG server is started, several server jobs will start. As clients sign on, they are assigned to these server jobs (one server will be assigned to its maximum before users are assigned to the next server).

     It is unlikely that this parameter would need to be changed. Increasing this parameter may be recommended if there are hundreds of very casual users. Decreasing this parameter may be recommended if there are just several heavy users.

   - MAXFRAME (on LIND): The maximum frame size parameter can be increased from 1994 bytes for TRLAN (or other values for other protocols) to its maximum of 16393 to allow for larger transmissions.

   - MTU size (CFGTCP command): The maximum transmission unit size parameter for both the route and interface effect the actual size of the line flows. Changing these values from 576 bytes to a larger size (perhaps the maximum of 16388) may reduce the overall number of transmissions. This needs to be considered on both the server and the Web browser. Typically, even simple screens have several thousand bytes of HTML data.

4. **WSG Memory Requirements:** Follow the faulting threshold guidelines suggested in the work management guide by adjusting the memory in the machine pool, the pool that the WSG servers run in, and the pool that the user jobs run in (use the WRKSYSSTS command). For the WSG server pool, the actual memory required per server (working set size) will vary significantly by the workstation I/O characteristics of the workload. From one benchmark measured in the lab, the memory requirement for the WSG server pool is:

600 K bytes + 700 K bytes for each WSG server

5. **Communications Lines and IOPs:** WSG/HTML sends significantly more bytes to the client workstation than Telnet would for the same application. Therefore, plan for additional communications line capacity to handle this extra load. For example, a Telnet transaction may consist of 1K bytes, while a WSG transaction may be 6K bytes for the same transaction.

6. **AS/400 Model Selection:** Use the information provided here along with the characterization of your WSG workload environment in a capacity planning exercise (perhaps with BEST/1).

   AS/400e servers or AS/400 Advanced Servers may not be the best choice for WSG environments. The user job is labeled as an ′interactive′ job, while the other tasks involved (WSG servers, comm tasks) all are tagged ′non-interactive′. For a typical commercial transaction with WSG, the ′interactive′ portion of the transaction is about 20% of the total transaction′s CPU time. To avoid decreasing the effectiveness of these models when using WSG, it is recommended that you keep the ′interactive′ portion of your overall CPU utilization below the threshold at where the efficiency drops off.

   - do not do heavy applications via WSG
   - limit WSG to casual use (low number of WSG transactions)

   Otherwise use traditional (non-server) AS/400 models.

6.3 Firewall Performance

Using the Integrated PC Server as a firewall provides additional value-add for the AS/400 as a web server. The firewall can be on the same system as the web server or on a different system within the network. With the Integrated PC Server handling the firewall activity, the AS/400 CPU is not significantly impacted.

″Web server behind the firewall″: In this scenario the Integrated PC Server is performing packet filtering and allows HTTP traffic only through to the web server (also on the same AS/400). For an optimally configured system, having the firewall function active under a load, only slightly degrades the overall AS/400 web server capacity (compared with a similar, non-firewall configuration).

If a system is not optimally configured the decrease can be more significant. For example, if the MTU size is reduced to 500 bytes, then the impact of the firewall can be a 50% capacity reduction.

Chapter 7. Java Performance

Highlights:

- Introduction
- Improved Computational Performance (software and hardware)
- Reduced main storage footprint
- JDBC Improvements and Commercial Java Applications
- Unmatched Scalability
- Comparison to Existing Languages
- Java Performance -- Tips and Techniques
- Recommended Models and Capacity Planning

In traditional AS/400 applications, the performance of the application program itself is often a small contributor to overall performance. A large percentage of the execution is system services (e.g. Data Base Get Records) used by the application. Two ways to improve application performance are: 1) IBM improving OS/400, 2) The customer improving how the application uses the system services (especially data base) in OS/400.

Java, as part of its portability story, will have a higher percentage of the application′s execution in Java programs and use less of a given Operating Service′s function. Accordingly, the computational performance of the Java language is more important. The significance here is that OS/400′s Java Virtual Machine (and the Direct Execution facility that translates Java bytecodes into directly executable programs) will be more important than in other languages. It will also be true that inefficient coding practices in the application and any third party Java code will also be more evident compared to traditional applications.

We′ll discuss both computational performance (which is often measured by noncommercial programs) and commercial performance (Java implementing things recognizably commercial).

Because there are no rigorous, public benchmarks, we′ll use our own internal work. We believe that the magnitudes shown are a good guide to what customers will see when using our product.

7.1 Improved Computational Performance

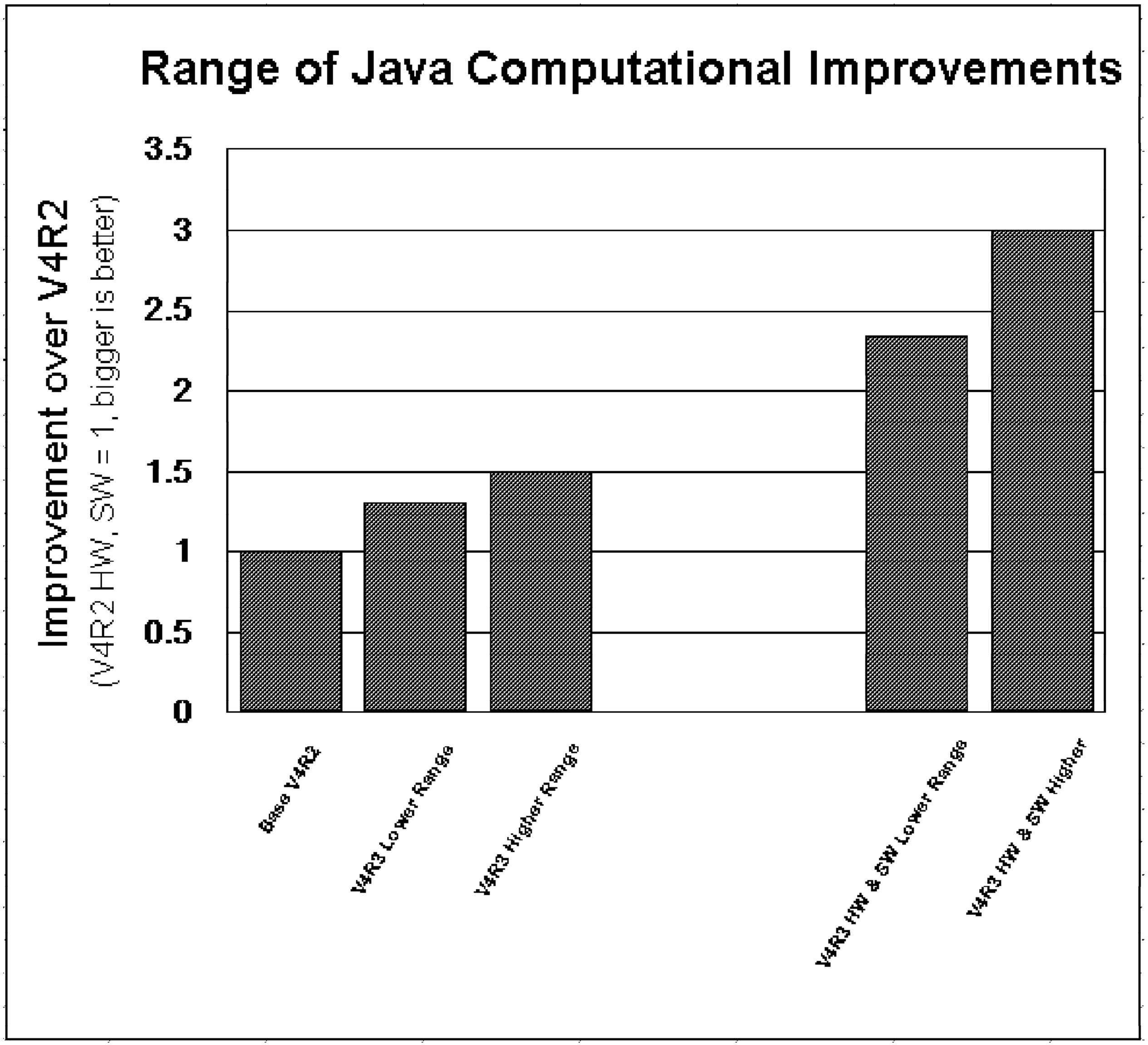


Figure 23. Java Computational Improvment, V4R3

Software Improvements

AS/400′s Java product features improved computational performance over V4R2. In the chart above, the second two bars from the left show the expected range of improvements over V4R2.

Internal benchmarks suggest that the exact amount of improvement can vary greatly. However, a typical Java application, with minimal data base interactions, will improve approximately thirty to fifty percent over V4R2, on the same hardware.

This improvement comes about because of improvements in both code generation and in the allocation and garbage collection strategy of AS/400 Java′s virtual machine. Particularly in the latter case, widespread improvement is likely for typical Java server applications over V4R2.

Hardware Improvements

The V4R3 release of AS/400 features the introduction of a new processor family to existing model numbers. As their CPW ratings indicate, these features have improved hardware performance.

In our internal Java benchmarks, we are seeing 1.8 to 2.0 times the performance of the corresponding model of the V4R1 generation of hardware, when both machines are running V4R3. Thus, the improvement can approach 100%.

Joint Hardware and Software Improvement

For those customers who upgrade both their operating system to V4R3 and who also using V4R3 hardware in place of the earlier V4R1 model, substantial gains should be possible. The expected improvement range is shown in the rightmost two bars in the above chart. In common computational scenarios, single job processor performance is the main issue, so scalability questions don′t arise.

Thus, the improvement range for those who upgrade both their hardware and software is from 2.3 to 3.0 times their current performance, as measured by a single job′s performance. That′s up to a 200% increase.

For commercially oriented applications, on AS/400, see the later JDBC discussion (the outcome is similar, but the reasons are different).

7.2 Reduced Main Storage Footprint

Given the current state of the art, accurate calibration of the main storage footprint on any machine is difficult. But, customers who experiment with V4R3 applications′ performance versus their V4R2 applications should see improved performance in smaller storage pools in most cases. Because it is possible to increase processor consumption to reduce main storage, and vice-versa, and because that trade-off is application dependent, exact predictions cannot be made. Customers might try experiments where they reduce the pool size bit by bit, and observe the effects on Non Data Base Paging on the WRKSYSSTS display. (This sort of investigation should not be done on a mission critical machine). In many cases, the reduction in pool size should have little effect for a while, until some application-dependent threshhold is reached where the processor performance for reducing the next increment of storage has too high a penalty.

7.3 JDBC Improvements and Commercial Java Applications

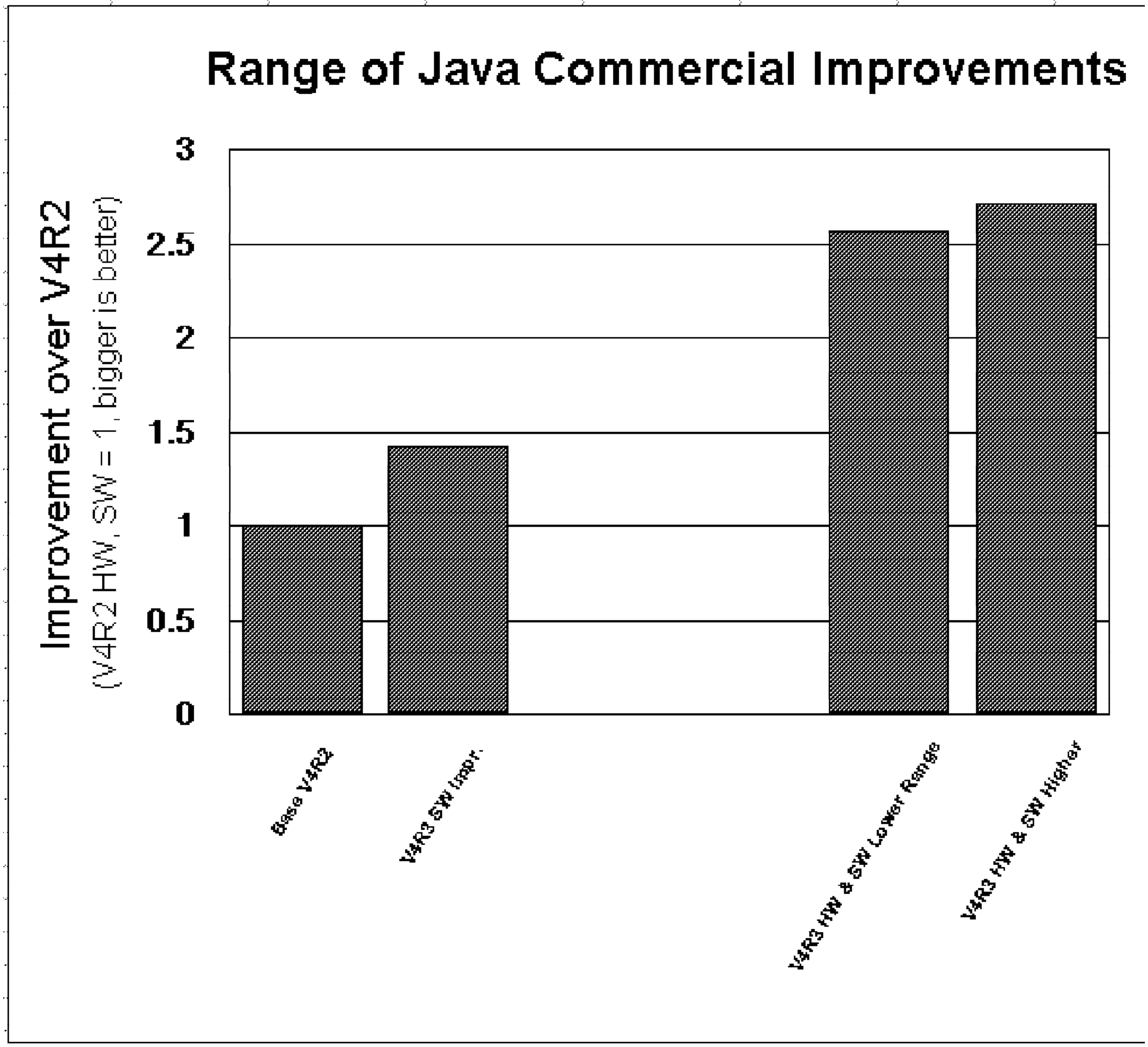


Figure 24. Java Commercial Improvment, V4R3

One key and frequent distinguishing mark between strictly computational programs and commercial programs is use of a data base, usually intense use of a data base.

Because of improvements in the JDBC driver logic, we expect data base intensive applications to see substantial improvements in many cases. Our internal commercial benchmark, which features extensive JDBC data base and multiple jobs and threads, is designed to demonstrate both basic commercial performance and scalability.

On the performance side, our internal application saw an improvement due to software of about 40 percent over V4R2. The second two bars from the left on the chart above illustrate this. Much of this improvement comes due to improvements in the JDBC driver performance.

For those who would have purchased the current generation of hardware, but now purchase a V4R3 model, a similar range of improvement was observed in hardware as was the case for computational Java. See the rightmost two bars on the above chart for the joint improvment of hardware and software.

Collectively, then, improvements from 2.5 to 2.7 should frequently be seen when upgrading both the hardware and software to V4R3. This is an increase of 170% over existing application performance.

7.4 Unmatched Scalability

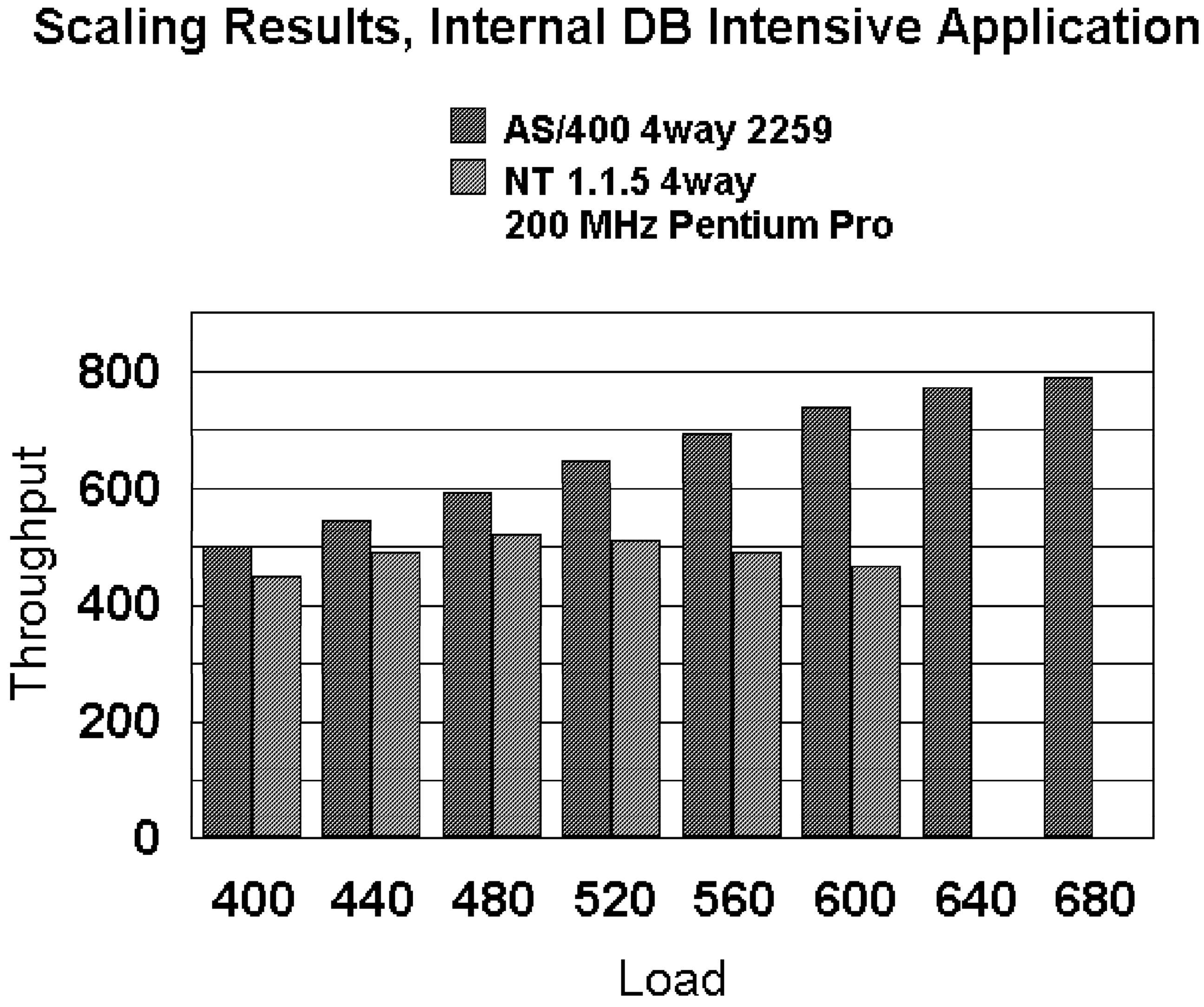


Figure 25. Java Scaling Story, V4R3

AS/400 Java maintains its scalability advantages over the competition in our internal benchmarks. Typically, we see that competitive JVMs (especially NT-based Java Virtual Machines) tend to ″run out of gas″ as the workload increases.

Especially when the configurations are larger, therefore, an AS/400 will be an excellent choice because it will be able to scale farther and better that competitive offerings.

There are cases where the NT implementation will outperform AS/400 at smaller loads and industry publications are sometimes based on these smaller, briefer, and easier-to-run evaluations. However, when these are scaled up, sometimes as little as by a factor of 10, AS/400′s scalability advantages begin to show up.

The only platform we know of with a comparable scalability story is RS/6000 under AIX. But, AS/400 exceeds RS/6000 in scalability.

7.5 Comparison to Existing Languages

Despite improvements, Java′s performance continues to lag more traditional languages in performance. This is true in both the computational and the commercial (data base) application arena. Reasons for this continue to be Java′s relative immaturity, and some common Java programming idioms which ease programming (especially for advanced functions like multi-threading) but hurt performance. See ″Tips and Techniques″ below for ways to minimize these problems. Before going further, we should note that all competitors we are aware of share this performance deficit. We should also note that improvements in Java performance in AS/400 as well as other platforms, are larger than improvements seen in traditional languages.

Java continues to narrow the performance gap, compared to more mature languages. Moreover, it is approaching the two-to-one performance boundary. History suggests that when a programming revolution reaches the two-to-one performance boundary (be it Java versus C++, C++ versus C, or even FORTRAN versus Assembler), the newer programming paradigm takes over due to its productivity and other benefits. History also says performance eventually approaches parity thereafter. Java is closing in on that threshhold, though it hasn′t reached it yet.

That said, industry projections showing Java already at or near parity with other languages remain largely hype. We do find the occasional program where Java can outmatch other languages. However, we know of no suite of programs making up any of the Industry Benchmarks where anyone′s Java has yet reached parity with what C, C++, or RPG could provide.

Internal work has confirmed that typical computational Java programs will lag traditional languages. Typically, Java is slower by factors ranging from 2.5 to 4. We have confirmed this result two ways: 1) By comparing computational programs written in both C and Java, and 2) By comparing simple functions written in both RPG and Java. The RPG comparison includes data base operations. The results are comparable and in the 2.5 to 4 range.

Java also lags in commercial performance. Java (via JDBC) uses dynamic SQL and extended dynamic SQL. SQL is very general and functionally rich, but has a certain overhead, especially for simpler functions. Traditional COBOL and RPG use simpler, highly tuned native data base access. Accordingly, the JDBC improvements, while welcome, will not reach parity with traditional languages. For Java programmers willing to bypass a general Java solution and use native AS/400 data base, the AS/400 Toolbox for Java offers alternative methods for achieving better data base performance. That said, we expect most developers to use the standard JDBC access methods because it is part of the Java portability story. Like the rest of the industry, we will continue to look for ways to improve JDBC performance.

Java′s strengths in the V4R3 timeframe will continue to be:

- Multi-platform vendors who wish to exploit the ″write once, run anywhere″ property and sell their applications on AS/400 as well as other Java machines
- Those needing simplified access to advanced functions such as networking, multi-threading, client-server, and object-oriented programming,
- Those who wish to web-enable variations of existing applications (e.g. order entry becomes customer order via web),
- Those looking for simple, productive access to SQL function,
- Those needing to bridge from the ASCII world (either from IFS files or the network) to EBCDIC data bases.

Customers needing maximum application performance should look at traditional languages like COBOL or RPG. C++ is also available on AS/400, providing higher performance for Object-Oriented applications, but at a cost of greater complexity for programmers.

Some Java programmers may find that most of their performance problems occur on only a couple of highly used routines. For applications of this type, the Java Native Interface (JNI) allows the high use routines to be written in C or C++, but interoperate with the rest of the application in Java. If carefully limited to a few critical routines, better performance is gained without giving up Java′s other advantages.

7.6 Java Performance -- Tips and Techniques

Introduction

Tips and techniques for Java fall into several basic categories:

1. AS/400 Specific. These should be checked out first to ensure you are getting all you should be from your AS/400 Java application.
2. Java Language Specific. Coding tricks that will ordinarily improve any Java application, or especially improve it on AS/400.
3. Data Base Specific. Use of data base can invoke significant pathlength in OS/400. Invoking it efficiently can maximize the performance and value of a Java application.
4. Garbage Collection and Allocation Specific. Because Java programmers don′t directly return their unused storage for reuse, the Java garbage collection facility must run on occassion to claim unused storage. Tuning the execution of garbage collection can be highly important to performance. This can be done by tuning garbage collection′s performance or by avoiding the creation of new objects (see also language specific suggestions).

AS/400 Specific Java Tips and Techniques

- *Load the latest CUM package and PTFs*
  To be sure that you have the best performing code, be sure to load the latest CUM packages and PTFs for all products that you are using. Information on the AS/400 JVM can be found at the ″http://www.softmall.ibm.com/as400/java/devkit.html″ Developer Kit for Java Web Site. Information on the AS/400 Toolbox for Java can be found at the ″http://www.as400.ibm.com/toolbox″ Toolbox Web Site.

- *Use CRTJVAPGM on .class files.*

  Java .class files should be converted into direct execution (machine instruction) Java program objects through the CRTJVAPGM command. Use this command before running any Java programs. The Program object is permanent and will be re-used once it is created. Use the DSPJVAPGM command to determine if the Java Program Object exists. Optimization levels 10, 20, 30 and 40 are supported on the CRTJVAPGM command. In most cases, optimization level 40 should be used. Opt 40 ordinarily gives the best performance.
  Relative Performance:
  Results will vary by application. For computation and call intensive applications the relative gains can be dramatic. Here are the relative performance gains for a well-known computational problem, ″The Towers of Hanoi.″

	Relative time (bigger is slower)
Optimization level 40	1.00
Optimization level 30	1.73
Optimization level 20	2.02
Optimization level 10	2.65
Interpretive	12.95

  Comparisons based on V4R2, JDK 1.1.4. Similar magnitudes would be observed on V4R3.

- *Use CRTJVAPGM on .zip and .jar files*
  CRTJVAPGM must be used on .zip and .jar files. The JAVA/RUNJVA command will implicitly create program objects for each class file in a .zip or .jar. Unlike a .class file however, these program objects are not saved. The implicit creation of program objects will significantly impact JVM startup or class loading. The implicit creation will occur every time you start the JVM. To determine if your .zip/.jar file has a permanent program object use the DSPJVAPGM command.
  NOTE 1: Packaging multiple classes in one .zip or .jar file may also improve class loading time and code optimization.
  NOTE 2: If using the AS/400 Toolbox for Java classes on v4r2, you will need to perform CRTJVAPGM on jt400.zip and jt400.jar.

Java Language Performance Tips

- *Minimize synchronized methods*

  Synchronized method calls take at least 10 times more processing than a non-synchronized method call. Synchronized methods are only necessary if you have multiple threads sharing and modifying the same object.
  If the object **never** changes after it is created (″constructed″ is the Java term for ″created″), you don′t need to synchronize any of its methods, even for multi-threading.

  Note: Dealing with synchronized methods mean understanding some important Java programming concepts.

  – Some Java objects, notably String, do not permit data in the object to be modified after the object is constructed. For such objects, synchronized methods are never needed.

  – Other objects, such as StringBuffer, allow the object to be modified after construction. All of its methods are synchronized.

  – Many objects fit these two models. If a StringBuffer type object will be used by **even one** multi-threaded application, all methods must be synchronized except its constructors. If you never use multi-threading, then a StringBuffer type object requires no synchronization.

  – But, consider object reuse when you decide. If some later application uses your object, and that new application is multi-threaded, synchronization will be needed. This is why common Java objects like StringBuffer have synchronized methods.

- *Use the -O javac option*

  The javac compiler may in-line methods if the -O option is selected. In order for a method to be in-lined it must be a final method (static, final, or private). Final methods can not be subclassed, which is the key to allowing in-lining. Method in-lining is not currently supported in the AS/400 JVM other than through the -O option.

- *Minimize object creation*

  Object creation can occur implicitly within the Java APIs that you use as well as within your program. Object creation and the resulting garbage collection can typically take 15% to 30% of a server transaction workload. To minimize this cost you can reuse an object′s space implementing a ″reset″ method that reinitializes the local variables in the object. The code fragment

  ```
  if objx = null then
     objx = new x(some,creation,parameters);
  else
     objx.reset(some,recreation,parameters);
  ```

  can provide significant performance improvements.

  Common causes of object creation that may not be obvious:

  – The I/O function readLine creates a new String.

  – Invoking the substring function of a String creates a new String.

  - The JDBC Result Set function getString creates a String.
  - The StringTokenizer returns a String from many functions.
  - Passing a scalar **int** or **long** as an object will create an Integer or Long object.

- *Minimize the use of String objects*
  String objects in Java are immutable. This means that you can not change (append, etc.) to a string object without creating a new object. Object creation is expensive and can occur multiple times for each String object you are using. To minimize the use of String objects you should use either StringBuffer or char•“. StringBuffer may also be a problem since the StringBuffer classes use synchronized method calls. An array of characters (char[]) can be used to simulate fixed length strings. This is recommended for applications which make heavy use of string data.
  Relative Performance:
  The following table shows the relative performance difference when using String, StringBuffer, or char[]. The test case concatenates two strings. For the char[] case. the concatentation reduces to simple array assignment, thus avoiding the creation of objects and the synchronization overhead associated with StringBuffer. In the following table the String ″World Wide″ was concatenated to the string ″Wait″. For the char[] case there were simply four char[] assigments for the characters ′W′ ′A′ ′I′ ′T′.

	Relative time (bigger is slower)
char[]	1
StringBuffer	35
String	363

Comparisons based on Optimization level 40, V4R2. JDK 1.1.4. V4R3 results would be similar.

- *Leverage variable scoping*
  Java supports multiple techniques for accessing variables. One typical technique is to write an ″accessor″ method. Local variables are the fastest.
  Relative performance:
  Here are five comparisons on variable access time and their relative performance. A local variable is the fastest and is given a relative performance of 1.

	Relative time (bigger is slower)
Local variable	1.0
Instance variable:	2.0
Accessor method in-lined:	2.0
Accessor method:	5.9
Synchronized accessor method:	99.6

Comparisons based on Optimization level 40, v4r2. JDK 1.1.4. V4R3 relative differences would be comparable.

Note: This is a performance-oriented suggestion. Making instance variables public reduces the benfit of Object Orientation. Having a local copy in the method of an instance variable can improve performance, but may also add coding complexity (especially in cases where individual blocks use the synchronized keyword). Avoiding the ″synchronized″ label on a method just for performance may lead to difficult bugs in multi-threaded applications.

- *Minimize use of exceptions (try catch blocks)*
  Exception handling requires a certain amount of ″setup″ information. A ″try″ statement in Java results in some setup so that if an exception occurs, the ″catch″ routine will get control. This adds pathlength.

  Sometimes, exceptions are required by a given method on a given object. These must obviously be coded. But, try and avoid having too many of these.

- *Do not start up too many JVMs*
  The JAVA/RUNJVA commands create a new batch immediate Job to run the JVM. Limit this operation to relatively long running Java programs. If you need to invoke Java frequently from non-Java programs consider passing messages through an AS/400 Data Queue. The ToolBox Data Queue classes may be used to implement ″hot″ JVM′s.

- *Use static final when creating constants*
  When data is invariant, declare it as static final. For example here are two array initializations:

```
class test1 {
     int myarray[] =
              { 1,2,3,4,5,6,7,8,9,10,
                     2,3,4,5,6,7,8,9,10,11,
                     3,4,5,6,7,8,9,10,11,12,
                     4,5,6,7,8,9,10,11,12,13,
                     5,6,7,8,9,10,11,12,13,14 };
     {

class test2 {
     static final int myarray2[] =
               { 1,2,3,4,5,6,7,8,9,10,
                                   2,3,4,5,6,7,8,9,10,11,
                                   3,4,5,6,7,8,9,10,11,12,
                                   4,5,6,7,8,9,10,11,12,13,
     }                             5,6,7,8,9,10,11,12,13,14 };
```

Relative Performance:
When 10,000 objects of type test1 and test2 were created the relative time for test1 was about 2.3x longer than test2. Since the array myarray2 in class test2 is defined as static final, there is only ONE myarray2 array for all 10,000 instances of test2. In the case of the test1 class, there is an array myarray for each test1 instance.

Comparisons based on Optimization level 40, v4r2. JDK 1.1.4.

AS/400 Database Access Tips

- *Use the native JDBC driver*
  There are two AS/400 JDBC drivers that may be used to access local data. Programmers coding connect statements should know that the Toolbox driver is located at Java URL *″jdbc:as400:system-name″* where system-name is the AS/400 TCP/IP system name. The native JDBC driver is located at Java URL *″jdbc:db2:system-name″* where the system-name is the Data Base name. The native AS/400 JDBC driver uses an internal shared memory condition variable to communicate with the SQL/CLI Server Job. The ToolBox JDBC driver assumes that the data is remote and uses a socket connection into the client access ODBC driver. The native JDBC driver is faster when you are accessing local data.

- *Use Prepared Statements*
  The JDBC prepareStatement should be used for repeatable executeQuery or executeUpdate methods. If prepareStatement is not used, the execute statement will cause an implicit prepareStatement everytime the execute statement is run.

  Note: Avoid placing the prepareStatement inside of loops (e.g. just before the execute). In some non AS/400 environments, this just-before-the-loop coding practice is common for non Java langauges. Programmers may carry it over to Java. However, in many cases, the prepareStatement doesn′t change and the Java code will run faster on all platforms if it is executed only one time, instead of once per loop. It will show a greater improvement in AS/400.

- *Store character data in DB2 as Unicode*
  The AS/400 JVM stores string data internally as 2 byte Unicode. If you are reading or writing large amounts of string data and the data is stored in EBCDIC, the data will be converted on every database access. You can avoid this conversion by storing the data in DB2 as 2 byte Unicode. Use the SQL graphic type with CCSID 13488 or the DDS graphic type with CCSID 13488.

  Note: Be careful with this suggestion. 1) If characters are the main portion of the record data, the record could double in size. If this is a large and important data base, this will increase hard disk expense, perhaps by a large amount. 2) If the data base is accessed by non Java code (e.g. legacy RPG applications) Unicode may create complications for the old code.

- *Store numeric data in DB2 as double*
  Decimal data cannot be represented in Java as a primitive type. Decimal data is converted to the class Java.lang.BigDecimal on every database read or write. This conversion can be avoided by storing numberic data in the database as float or double. Alternatively, you may considering converting the BigDecimal type to double for heavy calculations while leaving the Database in decimal form. Note that rounding differences may be introduced with the use of double. Float can also be used, but most C programmers use double for this purpose to minimize rounding difficulties.

- *Use ToolBox record I/O*
  The AS/400 ToolBox for Java provides native record level access classes. These classes are specific to the AS/400 platform. They provide a significant performance gain over the the use of JDBC access.

- *Check ToolBox for existence of a Java program object*
  The jt400.jar file contains the AS/400 ToolBox for Java product. After installation, this .jar file does not have a Java program object. Use the

CRTJVAPGM at optimization level 40 to create the program object. Use the CRTJVAPGM command during low system activity as it will take some time. Use the DSPJVAPGM command to see if the program object already exists.

Allocation and Garbage Collection

- *Leave GCHMAX as default*
  The GCHMAX parameter on the JAVA/RUNJVA command specifies the maximum amount of storage that you want to allocate to the garbage collection heap. In general the default value (Set to the largest possible value) should be used. The system does not allocate the storage until it is needed. A large value does not impact performance. If this maximum is reached, the JVM will stop all threads and attempt to synchronously collect objects. If GCHMAX is too small, a java.lang.OutOfMemory error will occur.

- *Adjust GCHINL as necessary*
  The GCHINL parameter on the JAVA/RUNJVA command specifies the amount of initial storage that you want to allocate to the garbage collection heap. This parameter indirectly affects the frequency of the asynchronous garbage collection processing. When the total allocation for new objects reaches this value, asynchronous collection is started. A larger value for this parameter will cause the garbage collector to run less frequently, but will also allocate a larger heap. The best value for this parameter will depend on the number, size, and lifetime of objects in your application as well as the amount of memory available to your application. Use of OPTION(*VERBOSEGC) can give you details on the frequency of garbage collection, and also object allocation information.

- *Ignore GCHPTY*
  This parameter is not used. It has no effect on performance.

- *Ignore GCHFRQ*
  This parameter is not used. It has no effect on performance.

- *Monitor GC Heap faulting*
  Java objects are maintained in the JVM heap. Excessive page faults may occur if the memory pool for your JVM is too small. These faults will be reported as non-database page faults on the WRKSYSSTS command display. Typically, the storage pool for your JVM is *BASE. Fault rates between 20 and 30 per second are acceptable. Higher rates should be reduced by increasing memory pool size. In some cases, reducing this value below 20 or 30 per second may improve performance as well. If you have the storage available, reducing the rate below 20 to 30 per second may be a benefit. Lowering the GCHINL paramater might also reduce paging rates by reducing the AS/400 JVM heap size.

- *Minimize Object Creation.*
  See previous suggestion about minimizing object creation.

7.7 Capacity Planning and Model Recommendations

Java represents a new set of trade-offs for the IBM AS/400 community. Since a product line cannot have separate prices and models for every technology, adapting our existing BEST/1 and CPW information is the best choice for configuring and explaining Java proposals for customers. This requires some reflection and work when recommending machines with Java content.

While other sources exist (look around http://ca-web.ibm.com), following may be used to bound your recommendations about Java deployment for customers:

First, remember to account for the amount of traditional processing (RPG, COBOL, etc.) going on in your customer′s shop. To the extent this is going on, the machine should be sized according to existing capacity planning guidelines. This also applies to customers using newer functions like SAP and Lotus whose characteristics are well-known.

Second, be careful to ascertain how much Java is going into the AS/400 itself. If AS/400 is being accessed by client Java code, but the code in the AS/400 itself remains in COBOL, RPG, or C/C++, there′s no point in padding capacity for Java -- it isn′t being used. The important item in that case becomes ensuring the customer′s PCs have enough performance to run Java on the client, and enough traditional horsepower to service their requests. Likewise, if the AS/400 is just being used as a Web Server (e.g. Domino GO), there′s no need to change capacity planning for Java content for that reason alone. Until Java is used in CGI-BIN function or a Java application, an AS/400 running a Web Server will not itself be running Java function.

Third, recognize that AS/400 has been optimized around scaleable, OLTP type applications which use lots of system services such as data base. *Java, by contrast, will tend to put more of its execution in the application itself.* The reason relates to Java′s portability story. This property will tend to increase processor requirements overall compared to what we′re familiar with. How much the increase would be varies by application, but Java applications will usually desire more processor resources than traditional ones. They will also need more main storage, in part because of tradeoffs in the language that will increase storage requirements versus traditional languages.

Fourth, because of the increased processor needs, be wary about recommending the smallest AS/400 models. Look for upgrade opportunities. This is particularly true of test and development machines. Because of OLTP price/performance tradeoffs, smaller or older machines may be disproportionately disappointing to customers when used for Java, even for testing. In general, make sure the processor performance of the test machine and development machines is in line with that of the production machine. This would mean using a machine with a higher Uniprocessor CPW rating than would ordinarily be the case.

Fifth, be aware that customers will be willing and able to write their own benchmarks for Java. Most of these will not play to AS/400′s strengths (multiple jobs, multiple threads) and so we may tend to look disproportionately bad. If customers have such a benchmark, work with them. Make sure that it: a) is some kind of prototype of a true ′server′ application, b) runs long enough (at least 15 minutes) to represent a fair, steady-state comparision, c) has scalability characteristics (multiple threads, multiple Java jobs, etc.).

Sixth, recognize that Java won't deploy in a traditional manner. 5250 operations to and from Java will not be a frequent attribute of Java on the AS/400. Accordingly, **the higher the Java content, as a percentage of total operations, the more a server model should be considered.**

Seventh, keep in mind that not everything changes for Java.

- Whether or not a multi-processor(SMP) is a good idea or not will seldom change for a Java-based application versus a traditional application. The only exception might be if the customer is going to use multiple threads where the RPG would have been a single job; if the RPG would use multiple jobs, evaluate SMP as usual.
- The size of the data base should ordinarily be identical in size, which ordinarily means disk requirements would be the same. Increases in disk might come if Unicode rather than EBCDIC is used for character data, or, pathologically, if a really massive amount of Java code came to the machine. In most cases, neither is likely enough to worry about. Ordinary considerations for disk capacity should apply very widely.
- In some cases, because Java will trail traditional languages in processor performance, a reduction in disk capacity (compared to traditional) might occur. However, an assumption of disk parity is ordinarily wise, especially if the machine has a large number of non-Java applications, new or old.

Better Machines for Java

Relative to their CPW rating, the following are better choices for Java:

AS/400 Model	*Processor Feature Number*
650	2189, 2188, 2243, 2240
640	2239, 2238, 2237
620	2182, 2181
530	2162, 2153, 2152, 2151, 2150
S40	2189, 2188, 2261, 2256
S30	2260, 2259, 2258, 2257
S20	2166, 2165
53S	2157, 2156, 2155, 2154
170	2386, 2385, 2292, 2291, 2290, 2183, 2176

Note that this list includes all models introduced in V4R3.

Machines to be used with caution with Java

Relative to their CPW rating, the following machines are to be used with caution when used for Java:

AS/400 Model	*Processor Feature Number*
620	2180, 2179,2175
S20	2163, 2161
170	2160, 2159

The 170, 620 and S20 models as a whole are more usable for Java than the 150, 600, 510, 500, 400, and S10 models. You′ll see ″higher end″ type numbers of 170, 620, and S20 on the ″better″ list.

Machines not recommended for Java

The 150, 600, 510, 500, 400, and S10 line represent older AS/400 processors. Java was built with a newer and faster world in mind than these boxes provide. Relative to their CPW rating, the following machines are more likely to cause customer dissatisfaction when used for Java:

AS/400 Model	*Processor Feature Number*
600	2136, 2135, 2134, 2129
510	2144, 2143
500	2142, 2141, 2140
400	2133, 2132, 2131, 2130
S10	2119, 2118
50S	2122, 2121, 2120
40S	2112, 2111, 2110, 2109
150	2270, 2269

Chapter 8. IBM Network Station Performance

Performance information for Releases 1 to 3 for IBM Network Stations attached to V3R2, V3R7, V4R1 and V4R2 is included below. The following IBM Network Station functions are included:

- Time to initialize the IBM Network Station (prior to login) for ethernet, token ring and twinax
- Time to load the applications (5250 emulation and browser, etc.)
- 5250 application performance
- Browser performance
- Java Virtual Machine (VM) applet/application performance
- Times for the IBM Network Station series 100, 300 and 1000

The computer industry has a generic name for the IBM Network Station - the thin client. Since clients attach to servers, it might seem that an AS/400 SERVER model attached to a Network Station (a thin CLIENT) would always be the best fit (that is, CLIENT to SERVER). Be cautious when the Network Station is attached to an AS/400 SERVER model. When using 5250 applications, the Network Station looks like a Non-Programmable Terminal (NPT) (like an interactive job) to the SERVER and will be subject to the AS/400 SERVER interactive rules, so it might not always be a good fit. The traditional AS/400 SYSTEM models are always a good fit with the Network Station.

8.1 IBM Network Station Network Data

The following table shows the amount of data that flows from the AS/400 to each IBM Network Station for initialization and each application load:

Table 22. Elements Loaded to a Network Station (MB)

Release	**Rel 1-2.5**	**Rel 1-2.5**	**Rel 3**	**Rel 3 DBCS***
Series	100/300	1000	All	All
Kernel + Configuration + Other	4.0	4.8	3.0	3.9
5250 Emulation	0.9	0.9	1.6	3.8
3270 Emulation	0.3	0.3	0.9	3.2
IBM Network Station Browser	2.2	2.2	NA	NA
Navio NC Navigator	3.7	3.7	5.0	10.0
Java Virtual Machine	1.5 - 5.0	1.5 - 5.0	1.5 - 5.0	1.5 - 5.0

Note:
- the amount of data downloaded will vary, depending on the configuration selected
- *DBCS support includes Japanese, Korean, simplified Chinese and Traditional Chinese

The kernel/configuration data is downloaded when the Network Station is powered-on. Unless configured otherwise, all the other options are downloaded when they are selected.

Note that when an application (e.g. 5250 emulation) is closed or the user logs-out, that application will again be downloaded when it is next selected - it is not kept in memory across log-outs. The kernel/configuration data is kept in the Network Station across log-outs.

The Java Virtual Machine download time varies depending on the application. Only the required classes are downloaded.

In Release 3, some of the information that is sent to the Network Station is compressed. Once received, the Network Station decompresses it. This compression means fewer bits are shipped from the AS/400 to the Network Station, resulting in better LAN utilization. More data/function is shipped to the Network Station in Release 3 than in previous releases. The compression results in boot performance that is about equal to previous releases.

Release 3 contains an option, TFTP subnet broadcast, that can significantly decrease the amount of data transmitted during the boot process, as well as saving significant CPU cycles in the AS/400. This option is described further in the sections below.

8.2 IBM Network Station Initialization

Initialization, at this time, is not trivial and could be a performance concern for some customers. The time to initialize the Network Station, particularly when many stations are initialized simultaneously can be prohibitive. In addition, initialization can consume a lot of AS/400 CPU, so that other jobs on the AS/400 might be starved.

If possible, it is best to leave the Network Station powered-on after initialization and/or to stagger initialization. The IBM Network Station consumes very little power. If initialization times are a problem and power outages are a concern, battery backups for each IBM Network Station, should be considered or possibly server systems dedicated to initialization.

Different Initialization Mechanisms - the Gory Details

Initialization is performed using TCP/IP Trivial File Transfer Protocol (TFTP) and/or AS/400 Remote File System (RFS). Both these access methods read files from the AS/400 to the Network Station. For reliability and performance, both mechanisms subdivide files into blocks for sending, and then recombine them in the Network Station. The TFTP block size can be configured 512 thru 8192 bytes. For Releases 1-2.5, the RFS block size is fixed at 8192. For Release 3, the RFS block size is fixed at 8192 until the Login window and 32K thereafter. TFTP and/or RFS is used during initialization depending on the configured initialization options.

There are three possible ways to initialize the Network Station:

- NVRAM - the AS/400 and Network Station IP addresses and other information are configured in each Network Station. The Network Station sends a TFTP request to the configured server to begin initialization.

- BOOTP - the Network Station broadcasts to find a responding AS/400 server. The AS/400 server is previously configured with each Network Station′s IP address and other information. Once the server receives a broadcast from a Network Station, it begins the initialization.
- DHCP - the same as BOOTP except the AS/400 server contains a pool of Network Station IP addresses.

BOOTP or DHCP is the preferred method, for Releases 1-2.5. All methods are OK in Release 3.

For Releases 1-2.5, NVRAM uses TFTP to load the kernel/configuration files and, after login, uses RFS. For Release 3, NVRAM uses TFTP to load the kernel and RFS for all subsequent files.

BOOTP and DHCP use TFTP to load the kernel and then use RFS to load all subsequent files.

For Releases 1-2.5, the Network Station tries 10 times with a 5 second timeout to locate and read the kernel using TFTP. After 10 attempts, an error message is issued. For Release 3, the Network Station can be configured to try indefinitely to locate and read the kernel.

For Releases 1-2.5, if NVRAM is selected, the Network Station reads the configuration files using TFTP. The Network Station will try 10 times with a 3 second timeout to read each file. If unsuccessful, it will skip that file and then try to read the next file - which eventually results in an unsuccessful initialization. (RFS has an infinite retry and will not skip files.) From a reliability perspective, this makes NVRAM, for Releases 1-2.5, the least preferred booting mechanism.

Release 3 contains a new option - subnet broadcast. Subnet broadcast is supported on ethernet, token ring and twinax. When this option is selected, TFTP data (the kernel - about 2MB), is broadcast from the AS/400 server to any requesting Network Station. That is, the kernel is sent one time so that each Network Station receives it. When subnet broadcast is off, the kernel is sent individually to each Network Station, which means a lot more data on the LAN/twinax. The broadcast is only to a subnet (e.g. any Network Station on a single ring, such as 9.5.112.x). When Network Stations from different subnets request the kernel, the AS/400 provides a broadcast to each subnet. The data below shows that subnet broadcast uses less AS/400 CPU. Subnet broadcast is the preferred boot option (twinax has some special considerations mentioned below). There is a caution - some routers do not support broadcast and broadcast can cause other problems, if not configured properly.

Subnet broadcast is supported on twinax. Unlike ethernet and token ring, the twinax protocol does not support broadcast. What this means for twinax, is that, when subnet broadcast is selected, each frame is sent individually to each device. When all devices are expecting the broadcast, this option works well (meaning less AS/400 CPU is used). When all devices are not expecting the broadcast, this option is not great (more data on the twinax cable). The data below illustrates this. In general, customers should not use subnet broadcast for twinax.

Some customers who have Series 1000s have experienced performance problems. The Series 1000 supports both full duplex and half duplex. In general, the performance problem is caused by a configuration error. The Series 1000 tries to operate in full duplex mode, but a router or something else in the network supports only half duplex. The Series 1000 almost continuously runs into collisions on the Ethernet which results in extremely slow performance.

Some customers who have token ring network switches that pass 4K frames have experienced difficulties. The customers had set their LAN frame size/MTU to a value greater than 8K. In general, these customers used NVRAM - with default 1024 TFTP block size. Initialization works fine until login - where RFS takes over and uses 8K frames. The 8K frames do not pass through a 4K switch. Some solutions to this problem might be: configure the switch to allow 8K frames, replace the switch with a router, or configure the AS/400 LAN frame size/MTU to 4K (twinax is fixed at 4K).

If the network has no Domain Name Server (DNS), performance can be very slow, since 30 second DNS timeouts occur frequently. AS/400 V4R2 contains DNS support. If a customer does not wish to use a DNS, for Release 3, good performance is still possible by doing the following:

- CFGTCP, Change TCP/IP domain information (option 12), set search priority to *LOCAL
- CFGTCP, Work with TCP/IP host table entries (option 10), add the IP address and host name for the AS/400 and each Network Station
- IBM Network Station Manager, select Hardware, select Workstations, under Domain Name Server, set Update Network Station Manager DNS file

The initialization options described in this white paper will fit most customer environments. There are other variations that can occur. For example, if the customer chooses BOOTP and successfully loads the kernel, but for some reason RFS isn′t working properly, initialization will timeout on RFS and switch back to TFTP. Variations such as these are not described in this document. The BOOTP boot sequence is decribed in greater detail in the following section.

BOOTP Initialization

There are four steps in the BOOTP initialization process. To get a total initialization time, times from each of the following four steps should be added together:

1. Hardware Test

   The hardware test is just that - a memory test and other hardware tests to ensure that the hardware is operational. For the most part, the length of this test is determined by the amount of memory in the IBM Network Station.

Table 23. Time (seconds) to perform hardware test

Memory (MB)	Series 100	Series 300	Series 1000
8	15	14	-
16	18	18	-
32	24	22	10
48	30	26	-
64	36	31	13

2. Kernel/Configuration Initialization

In this step, the Network Station locates the AS/400 server, reads the kernel and configuration files, and then displays the login window.

The Network Station broadcasts a BOOTP request to locate the AS/400 server. Then the kernel (about 2MB) is read using the TFTP function of TCP/IP. And then configuration files are read using the Remote File System (RFS).

The time to load the kernel using TFTP is heavily dependent on:

- TFTP block size
- TCP/IP maximum transmission unit (MTU) size
- LAN line description frame size (fixed for twinax)
- TFTP subnet broadcast option
- number of TFTP jobs

The Network Station negotiates the TFTP block size with the AS/400. It can range from 512 to 8192 bytes. The Network Station default is 8192. In general, the Network Station uses the TFTP block size, MTU and frame size defined by the AS/400.

The AS/400 default TFTP block size is 1024. As will be seen in the following tables, best performance is obtained with a large TFTP block size (e.g. 8192). If the MTU or frame size are less than 8192 (e.g. Ethernet has a maximum frame size of 1492) performance can be enhanced by configuring the block size greater than the MTU/frame size. If the TFTP block size is greater than the MTU/frame size, TCP/IP fragments (subdivides) the TFTP blocks to fit into the MTU/frame size. The Network Station TCP/IP recombines the MTU/frames into TFTP blocks. This fragmentation provides better performance than setting the TFTP block size equal to the MTU/frame size. Users should be aware that some routers, switches and/or gateways do not support this fragmentation capability. Twinax MTU/frame size are fixed, so fragmentation does not apply to twinax attached Network Stations.

The number of TFTP jobs on the AS/400 is also a performance factor - the optimal number for a system with a single LAN IOP is about 6, the default. The TFTP jobs are a pool of AS/400 jobs that download the kernel to Network Stations. They are first come, first serve. If there are more Network Station requests than jobs, the excess are ignored (i.e. not queued). If a request is not satisfied, the Network Station, every 5 seconds, will repeat its request. In general, there should be 6 TFTP jobs for each LAN IOP that has attached Network Stations.

The following tables and figures show how the TFTP block size affects the kernel/configuration initialization time, for a few AS/400 system sizes. The tables also show what happens when 1, 10, 50, and 100 Network Stations simultaneously (e.g. after a power outage) request TFTP initialization. The times represent the number of seconds when the last Network Station completes its TFTP and RFS download.

The data in the following tables was obtained in a dedicated environment. That is, only BOOTP, TFTP and RFS were running on the AS/400 and there was no other load on the LAN. In each test case, the base pool was cleared before beginning the test.

Results listed here do not represent any particular customer environment. Actual performance may vary significantly from what is provided here.

Table 24. Kernel/Configuration - AS/400 F97 and Network Station 300

Kernel/Configuration Initialization Time in Seconds
(Average CPU Utilization in %)
AS/400 Model F97 (V3R2)
IBM Network Station Series 300 (Releases 1-2.5)
16Mb Token-Ring
8KB MTU/Frame Sizes, 6 TFTP Jobs
Vary TFTP Block Size

# NS	512	1024	2048	4096	8192
1	109 (5.0)	46 (5.5)	34 (4.2)	29 (2.9)	26 (2.6)
10	225 (27.0)	105 (31.0)	77 (22.6)	63 (17.1)	57 (12.2)
50	992 (32.8)	470 (41.7)	327 (30.8)	257 (24.0)	209 (20.1)
100	1885 (35.2)	890 (46.3)	624 (33.6)	503 (25.5)	395 (22.3)

Note:
- results may differ significantly from those listed here

Table 25. Kernel/Configuration - AS/400 150-2270 and Network Station 300

Kernel/Configuration Initialization Time in Seconds
(Average CPU Utilization in %)
AS/400 Model 150-2270 (V3R7)
IBM Network Station Series 300 (Releases 1-2.5)
16Mb Token-Ring MFIOP
8KB MTU/Frame Sizes, 6 TFTP Jobs
Vary TFTP Block Size

# NS	512	1024	2048	4096	8192
1	85 (23.3)	35 (28.6)	31 (22.0)	27 (16.3)	26 (14.8)
10	229 (87.8)	126 (82.2)	83 (72.9)	63 (63.4)	55 (53.6)
50	1065 (94.2)	565 (95.0)	347 (92.0)	234 (87.6)	193 (77.6)
100	2075 (97.5)	1119 (97.0)	682 (94.5)	448 (92.5)	352 (88.1)

Note:
- results may differ significantly from those listed here

Table 26. Kernel/Configuration - AS/400 510-2144 and Network Station 300

Kernel/Configuraton Initialization Time in Seconds
(Average CPU Utilization in %)
AS/400 Model 510-2144 (V3R7)
IBM Network Station Series 300 (Releases 1-2.5)
2619 16Mb Token-Ring LAN IOP
8KB MTU/Frame Sizes, 6 TFTP Jobs
Vary TFTP Block Size

# NS	512	1024	2048	4096	8192
1	71 (9.8)	59 (7.4)	52 (6.4)	46 (5.8)	43 (5.2)
10	169 (39.3)	117 (30.3)	81 (26.1)	65 (21.2)	62 (17.3)
50	790 (44.5)	451 (42.4)	361 (32.6)	265 (28.7)	209 (27.0)
100	1526 (47.3)	875 (45.2)	667 (35.7)	498 (31.7)	384 (30.5)

Note:
- results may differ significantly from those listed here

Table 27. Kernel/Configuration - AS/400 S30-2257 and Network Station 300

Kernel/Configuration Initialization Time in Seconds
(Average CPU Utilization in %)
AS/400 Model S30-2257 (V4R1)
IBM Network Station Series 300 (Releases 1-2.5)
2629 16Mb Token-Ring LAN IOP
8KB MTU/Frame Sizes, 6 TFTP Jobs
Vary TFTP Block Size

# NS	**512**	**1024**	**2048**	**4096**	**8192**
1	96 (1.8)	41 (4.3)	33 (4.5)	30 (3.7)	29 (3.6)
10	182 (14.4)	73 (16.4)	56 (12.7)	52 (8.9)	39 (8.5)
50	735 (18.9)	279 (24.9)	201 (20.2)	146 (17.5)	127 (15.3)
100	1382 (20.2)	513 (27.7)	357 (23.2)	272 (20.2)	244 (16.6)

Note:
- results may differ significantly from those listed here

Table 28. Kernel/Configuration - AS/400 400-2132 and Network Station 300

Kernel/Configuration Initialization Time in Seconds
(Average CPU Utilization in %)
AS/400 Model 400-2132 (V4R1)
IBM Network Station Series 300 (Releases 1-2.5)
2629 10Mb Ethernet LAN IOP
1KB MTU/Frame Sizes, 6 TFTP Jobs
Vary TFTP Block Size

# NS	**512**	**1024**	**2048**	**4096**	**8192**
1	76 (35.6)	53 (26.3)	45 (19.8)	39 (17.7)	34 (15.5)
10	280 (90.2)	167 (82.0)	110 (72.6)	83 (63.7)	67 (55.7)
50	1311 (97.5)	745 (93.8)	467 (88.6)	321 (82.1)	277 (69.3)
100	2591 (97.8)	1466 (96.9)	895 (93.4)	623 (86.7)	540 (73.1)

Note:
- results may differ significantly from those listed here

Table 29. Kernel/Configuration - AS/400 400-2132 and Network Station 300

Kernel/Configuration Initialization Time in Seconds
(Average CPU Utilization in %)
AS/400 Model 400-2132 (V4R2)
IBM Network Station Series 300 (Release 3)
2629 10Mb Ethernet LAN IOP
1KB MTU/Frame Sizes, 6 TFTP Jobs
Vary TFTP Block Size

# NS	**512**	**1024**	**2048**	**4096**	**8192**
1	64 (33.1)	51 (22.9)	46 (15.9)	42 (13.4)	40 (10.6)
10	345 (82.8)	200 (75.0)	122 (62.4)	89 (51.1)	72 (48.4)
50	1831 (93.3)	1109 (89.9)	533 (85.6)	334 (81.7)	274 (67.6)
100	3678 (93.9)	1985 (90.1)	1111 (88.6)	660 (85.2)	543 (72.1)

Note:
- results may differ significantly from those listed here

Table 30. Kernel/Configuration - AS/400 400-2132 and Network Station 300

Kernel/Configuration Initialization Time in Seconds
(Average CPU Utilization in %)
AS/400 Model 400-2132 (V4R2)
IBM Network Station Series 300 (Release 3)
All NSs attached to a single twinax adapter
6 TFTP Jobs
Vary Twinax Adapter type, subnet broadcast option and TFTP block size

# NS	**6050 w/o subnet** 8K TFTP	**6180 w subnet** 1K TFTP	**6180 w/o subnet** 1K TFTP	**6180 w subnet** 8K TFTP	**6180 w/o subnet** 8K TFTP
1	107 (8.5)	114 (22.0)	116 (22.1)	87 (14.5)	82 (13.3)
2	173	155	133	90	85
3	225	165	154	106	98
4	275	168	159	121	116
5	325	186	178	142	139
6	388	201	199	155	157
7	446 (16.4)	225 (33.6)	221 (70.0)	171 (28.8)	162 (37.5)

Note:
- results may differ significantly from those listed here

Note that subnet broadcast uses less AS/400 CPU. However, as discussed above, each twinax device on the subnet will get their own copy of the broadcast data, even if they didn′t request it, which would mean unwanted data on the twinax cable. In general, customers should not use twinax subnet broadcast. (Subnet broadcast should be used on LANs.)

In Table 9, the Network Stations were all chained to a single cable port. For the 6180 adapter, faster times could be obtained if the Network Stations were balanced across cable ports, that is, half on ports 0-3 and the other half on ports 4-7. For example, in the table above, 6 Network Stations with an 8K TFTP block size, without subnet broadcast, booted in 157 seconds. If they had been balanced, 3 on port 0 and 2 on port 4, the initialization time would have been 130 seconds, 17% faster.

When multiple Network Stations simultaneously request initialization, the completion time of each Network Station depends on the configured number of TFTP jobs. In general, assuming subnet broadcast is not being used, Network Stations will complete their initialization in groups of six (or the configured number of TFTP jobs).

If a Network Station has multiple paths, with the same network address, to an AS/400 (e.g. two IOPs that each have a path to the Network Station), unexpected results may occur. Whenever the AS/400 gets a request from a Network Station, it uses the default path to get back to the requesting station. The return route (and any subsequent request/replies) may be different from the original request. This implies that there is no value to add a second IOP with the same network address to gain additional TFTP performance.

TFTP jobs are assigned first come, first serve. There is no mechanism to allocate a TFTP job to a particular IOP. This implies that it is possible for Network Stations attached to one network to monopolize all the TFTP jobs until completion of the kernel download. Other IBM Network Stations maybe starved until a TFTP job is available.

3. Login

   Login is just that - the user enters his/her user-ID and password and then the desktop appears.

   The load times can be found in the table below.

4. Application Load

   Applications are loaded when their respective desktop buttons are selected. Load times vary by AS/400 machines size.

   Getting to a 5250 sign-on can require two steps - from the menu bar, select the 5250 button to get to the host name window and then enter the desired host name to get to the 5250 sign-on window. Most administrators use the Network Station Manager to configure for direct menu bar to 5250 sign-on.

   Getting to the browser is a single step - from the menu bar, select the browser button to get to the browser.

   Examples of load times can be found in the table below.

Table 31. Application load times

Load Times in seconds **AS/400 Models 150-2270 and 510-2144 (V3R7)** **IBM Network Station Series 100 and 300 (Releases 1-2.5)** **2619 16Mb Token-Ring LAN IOP**				
	2270 100	**2270 300**	**2144 100**	**2144 300**
Userid/password to desktop	10	10	15	11
5250 select to host name	9	6	10	7
Host name to 5250 login	6	6	12	11
Browser select to browser	33	16	41	22
Note: - results may differ significantly from those listed here				

Table 32. Application load times

<table>
<tr><td colspan="5"><b>Load Times in seconds
AS/400 Model 400-2132 (V4R2)
IBM Network Station Series 300 (Release 3)
eSuite is IBM Network Station 1000 (Release 3)
Twinax or Ethernet Adapter</b></td></tr>
<tr><th></th><th>6050</th><th>6180</th><th>2629</th><th>2629 DBCS*</th></tr>
<tr><td><b>Userid/password to desktop</b></td><td>30</td><td>27</td><td>18</td><td>23</td></tr>
<tr><td><b>5250 select to host name</b></td><td>57</td><td>33</td><td>10</td><td>19</td></tr>
<tr><td><b>Host name to 5250 login</b></td><td>15</td><td>21</td><td>12</td><td>14</td></tr>
<tr><td><b>Browser select to browser</b></td><td>169</td><td>131</td><td>41</td><td>52</td></tr>
<tr><td><b>eSuite select to eSuite</b></td><td>-</td><td>-</td><td>175</td><td>-</td></tr>
<tr><td colspan="5"><b>Note:</b>
- results may differ significantly from those listed here
- *DBCS support includes Japanese, Korean, simplified Chinese and Traditional Chinese</td></tr>
</table>

Another example of subnet broadcast: Assume 100 Series 300 Network Stations attached to an AS/400 V4R2 2132 via a single 10Mb ethernet segment. Assume the electricity on all 100 Networks Stations goes out and a while later comes back on. Assume the Network Stations all have the same memory size (e.x. 32MB) and identical monitors attached. It would be possible for all 100 to be at the Login window in 280 seconds (less than 5 minutes). The 280 seconds comes from: 21 seconds for hardware test, 30 seconds to load the kernel, and 229 seconds to load configuration files.

8.3 AS/400 5250 Applications

The Network Station user should see 5250 applications almost exactly as with NPT or PC terminals. However, the load on the AS/400 may be different. Network Stations use the AS/400 TCP/IP Telnet path. Telnet consumes 27% more CPU time per transaction than an NPT attached to a local twinax for a typical commercial workload. This yields a 20% capacity reduction over a twinax attached NPT. For comparison, a Client/Access PC using 5250 over SNA, when using the same workload, consumes 10% more CPU time per transaction than a local twinax attached NPT.

The implication is that customers migrating from local twinax attached NPTs to LAN attached Network Stations will probably use more CPU to run the same 5250 applications. Customers migrating from LAN attached SNA Client/Access PCs will also probably use more CPU. Customers migrating from LAN attached TCP Client/Access PCs should need no additional CPU capacity to run their 5250 applications.

8.4 Browser

In general, the Series 100, 300 and 1000 all perform equally well. Their performance should be comparable to that seen on a PC.

It is important that either socks or proxy are configured, but not both. Poor performance is seen when both are used.

Disk caching should never be used.

8.5 Java Virtual Machine Applets/Applications

Java is still evolving. As such, its use on a Network Station is also evolving. The Series 100 clearly should not be used for Java. The Series 300, while twice as fast as the 100, can be used for very limited Java applets. The Series 1000 is for Java; however, since Java has varied uses, customers are encouraged to test their Java applications on the Series 1000 before putting them into production.

8.6 The AS/400 as a Router

The AS/400 is a router (data passes though it) when twinax attached Network Stations send/receive data from the internet or other servers. At this time, limited performance data is available. The following two tables show results when data is read from an NT server through an AS/400 to a Network Station.

Table 33. LAN to LAN Thruput

LAN to LAN Thruput
AS/400 Model 400-2132 (V4R2)
IBM Network Station Series 300 (Release 3) via 10Mb Eth to AS/400
300MHz PC NT server via 16Mb TR to AS/400
2629 LAN IOPs, 15MB of data, 8K TFTP block

# NS	Time (sec)	AS/400 Util (%)	AS/400 Thruput (Kb/s)
1	44	11.2	340.9
2	48	16.9	625.0
3	57	18.1	789.5
4	71	25.7	845.1
5	90	24.4	833.3
10	158	29.9	949.4
15	232	34.9	969.8

Table 34. LAN to Twinax Thruput

LAN to Twinax Thruput
AS/400 Model 400-2132 (V4R2)
IBM Network Station Series 300 (Release 3) via twinax to AS/400
300MHz PC NT server via 16Mb TR to AS/400
2629 LAN IOPs, 6180 Twinax adapter, 2MB of data, 8K TFTP block

# NS	Time (sec)	AS/400 Util (%)	AS/400 Thruput (Kb/s)
1	33	9.9	70.1
2	48	9.9	96.4
3	109	10.3	63.7
4	127	10.5	72.9
5	150	11.1	77.1
6	213	11.0	65.2

8.7 Conclusions

The IBM Network Station provides for an excellent working environment.

- In general, the Network Station 1000 performs better than the 300 which performs better than the 100
- Initialization
  - The Network Station Series 1000 initialization time is about the same as the 300, except for hardware test, where the 1000 is faster. The 300 is faster than the 100.
  - If possible, customers should consider a boot server for each ring or ethernet.
  - For Releases 1-2.5, customers should use BOOTP or DHCP and not NVRAM. BOOTP and DHCP are faster and more reliable. For Release 3, all three initialization mechanisms are equal in reliability and performance. BOOTP is slightly (1-2 seconds) faster than DHCP.
  - The time to initialize Network Stations depends on many variables, such as size of AS/400, TFTP block size, number of attached IBM Network Stations, LAN utilization, CPU utilization, etc. Customers will need to evaluate their own needs. It is recommended that customers go slow in building their Network Station solutions.
  - Initialization time varies from AS/400 model to AS/400 model. In general, the larger the model, the better the performance. On larger models, the bottleneck is the LAN IOP and, on smaller models, the bottleneck is CPU and LAN IOP. The 2629 LAN IOP provides better performance than the 2619.
  - 10Mb Ethernet, 100Mb Ethernet and 16Mb token ring are about equal.
  - During initialization, CPU utilization can be quite high, especially on the smaller AS/400s, which will impact other jobs. In addition, TFTP requires more CPU than RFS.
  - Subnet broadcast can significantly reduce LAN traffic and AS/400 CPU utilization. Subnet broadcast is available with AS/400 V4R2 and Network Station Release 3. If possible, it is highly recommended that subnet

broadcast be used. In general, subnet broadcast is not advisable with twinax, except as discussed earlier.

  - The network administer should configure TCP/IP, LAN frame size and TFTP block size for best performance. In general, the larger the size, the better the performance.
  - For twinax, the 6180 adapter is significantly faster than the 6050. The 6180 is about equal to a 4Mb token ring.
  - There is no value to add a second IOP, with the same network address, to a LAN to get better initialization performance, since TFTP will select the path to be used. All Network Stations, from the same network, will use the TFTP selected path.
  - It is best to configure 6 TFTP jobs per LAN that has attached Network Stations. However, for systems that have multiple LANs, since there is no way, at this time, to dedicate a TFTP job to a particular LAN, initialization may not perform as well as desired.
  - In general V4R2 provides better performance than V4R1, which provides better performance than V3R7. V4R2 contains TCP/IP and IOP LAN enhancements. In some cases, customers will see substantial improvements in kernel/configuration initialization. These improvements, in general, will be visible when a single Network Station is initialized with a small TFTP block size. V4R2 contains RFS enhancements.
  - Release 3 boots about as fast as previous releases, even though more data and function are sent. Much of the data sent is compressed.
  - Switches, routers and gateways can cause problems. It is best to have a network administrator.
  - For 6180 twinax attached Network Stations, best performance is obtained if all Express Datastream enabled devices are on the same cable, excluding older, non-Express capable devices.
  - When Express devices are attached to a single workstation controller, best performance is obtained by load balancing those devices. That is, half the devices should be connected to cable ports 0-3, and the other half should be connected to ports 4-7.

- 5250 application performance on the AS/400
  - In general, the 100, 300 and 1000 all perform equally.
  - Customers migrating from LAN attached SNA Client/Access PCs will probably use more CPU (about 17%) to run the same 5250 applications.
  - Customers migrating from LAN attached TCP/IP Client/Access PCs will use about the same CPU to run the same 5250 applications.
  - Customers migrating from local twinax attached NPTs to IBM Network Stations will probably use more CPU (about 27%) to run the same 5250 applications.
- Browsers
  - In general, the 100, 300 and 1000 all perform equally.
  - Poor performance is obtained when both socks and proxy are configured. Only one should be used.
  - Never use disk caching.

- Java Virtual Machine
  - The Series 100 should not be used for Java.
  - The Series 300 can be used for limited, lightweight Java
  - The Series 1000 is for Java; however, since Java hasn′t fully matured and can be used for many, varied applications, customers should insure that their Java application and the 1000 are compatible.
- AS/400 as a Router
  - Limited data is available. A model 400-2132 is able to route about 970kb/s from one LAN to another and about 75Kb/s from a LAN to twinax.

Chapter 9. AS/400 File Serving Performance

For V4R2 a new section on ***NetServer*** has been added. NetServer is AS/400 Support For Windows Network Neighborhood. See 9.3, "NetServer File Serving Performance" on page 147.

This chapter was previously extended with two additional sections containing V4R1 specific information. Existing content has been left intact due to the mutual exclusivity of the V4R1 information and to provide a sense of continuity.

This chapter will focus on the following topics.

- File Serving Performance Positioning
- Client Access/400 File Serving with the Integrated File System (IFS) File Server/400
- IPCS (aka FSIOP) with LAN Server/400 File Serving
- Multimedia File Serving
- V4R1 Update: QLANSrv Save/Restore
- V4R1 Update: WarpServer on IPCS

For information on AS/400 versus Microsoft′s SNA Server Gateway, refer to section "Related Publications / Documents" on page vii for directions on how to access a presentation on this topic.

Please refer to Chapter 5, "Communications Performance" on page 79 for general performance considerations for different connectivity methods.

9.1 File Serving Performance Positioning

V4R2 Note - No figures or tables were added or updated to reflect V4R2 changes. When studying this section, please notice the specific release listed with each set of data.

In V4R2, performance enhancements were made to the Integrated File System (IFS). The V4R2 improvements affect the Root, QOpenSys, and User-Defined file systems. Other existing file systems (QSYS, QDLS, QOPT, ...) will experience the same level of performance as they did in V4R1.

Three functions LSEEK, READ, and WRITE are all faster in V4R2. Users of the LSEEK API should see an improvement in the range of 2x. The READ and WRITE speeds will vary with the size and usage of these functions. Pre-bring buffering schemes were improved, along with other changes, resulting in READS and WRITES being upto 2x faster. As the number of files being accessed and the size of these files increase, the degree of improvement decreases to the point where there might not be any noticeable change in performance.

V4R1 Note - WarpServer/400 is a technology update for LAN Server/400. It provides further improvements to this type of file serving by

- using newer version of OS/2 WarpServer software
- enabling the use of the higher speed Pentium-based IPCS (types 6616 and 2850).

For ease of reference, all pre-V4R1 information will remain unmodified (except for a few V4R1 notations or cautions). V4R1 WarpServer information will be

″added″ in a latter section within. In the following discussion, statements about/for LAN Server/400 will be applicable to WarpServer/400. **A different** approach was used for V4R1 - notably dropping the use of the BAPCo5 workload.

With the unique requirements in the area of PC file serving, it is important to provide an environment to meet the needs of the customer. For PC file serving on AS/400 there are basically two options: LAN Server/400 with the FSIOP, and Client Access/400. If your requirement is for ***high performance*** file serving from client workstations running DOS, Microsoft Windows**, or OS/2* attached via LAN to AS/400, then LAN Server/400 and the FSIOP is the best solution. LAN Server/400 provides competitive PC file serving performance comparable to the leading PC servers, and requires dramatically less AS/400 CPU resource than using Client Access/400 and the Integrated File System for file serving.

Where performance is not a key requirement in PC file serving, such as in the areas of casual file serving or client administration, file serving provided by Client Access/400 is sufficient. Client Access/400′s strength is in providing seamless integration of the PC desktop not only in the area of file serving, but also in areas such as database serving, print serving, AS/400 application access, and 5250 emulation.

When you require both high performance file serving and access to all the PC desktop services provided by Client Access/400, LAN Server/400 and Client Access/400 can be used concurrently on the client. The LAN Server/400 requester provides access to all data maintained by LAN Server/400. Client Access/400 provides access to other AS/400 resources such as the AS/400 database, byte stream data in the Integrated File System (IFS), and printers.

IFS integrates the existing AS/400 file systems (QSYS.LIB - libraries/objects, and QDLS - folders/documents/shared folders) into a single hierarchical name space together with the following file systems:

- ″Root″: OS/2 and DOS compatible
- QOpenSys: POSIX, XPG, and UNIX** compatible
- QLANSrv: LAN Server/400 data - OS/2 LAN Manager compatible

File Serving Workloads and Configurations

The following workload environments and client/server configurations will be used in this chapter to compare file serving performance.

- **BAPCo5 Workload Description**

  The BAPCo5 (Business Application Performance Corporation) workload represents a client/server environment in which PC users run the following five commercially available applications and access programs, batch files, and data files residing on the server. Some of the functions utilized by each application are listed.

  – **Harvard Graphics**: macros, create org chart, 3-D pie chart, import Lotus and Excel spreadsheets, plot, create text chart, slide show, print

  – **Paradox**: define databases in the file system, selective retrieval, report generation, print

  – **WordPerfect**: small and large documents, scroll, help feature, tables, headings, search and replace, edit

  - **Excel**: bivariant normal distribution, print product forecasting, create tables, cut/paste, data entry, 2-D and 3-D graphs, format, print tax forms, load forms, enter data, scroll, links, print preview, print
  - **CCMail**: electronic mail, selective retrieval, edit, send response, print

  The preceding applications and end-user functions were selected based on research on which applications and functions are most prevalent.

  The applications and the files are maintained on the networked server and are accessed from client machines running Win-OS/2. Each client machine runs an automated script (beginning with a different application) and runs one application at a time until the entire set of applications has been run. The automated scripts run with zero think times and thus generate file serving requests that would be representative of many clients. Print output is held in a print queue on each client.

  Note: The BAPCo5 workload is a modification of the BAPCo workload described in the *AS/400 Performance Capabilities Reference (V3R2), ZC41-8166*.

- **Low Level PC Primitives Workload Description**

  The following file operations are executed from a client ″C″ program to files that reside on a network drive.

Test case	Description
Open100	100 files 2 dir deep for Read (i:\dir1\dir2\test)
Close100	Close 100 files 2 dir deep
Read1MB	Read (i:\dir1\dir2\test.001) 1048576 bytes/16384 bytes per read
Write1MB	Write (i:\dir1\dir2\test.001) 1048576 bytes/16384 bytes per write
Create100	Create 100 files 2 dir deep (i:\dir1\dir2\test)
Delete100	Delete 100 files 2 dir deep (i:\dir1\dir2\test)
GetFilAtt100	Get Normal File Attributes by Name 100 times

- **Client and Server Configurations**

  The following configurations were used to collect the data for the file serving measurements in this chapter.

  Servers

  - AS/400 model 400-2131
    96 MB memory, 33000K for Machine Pool, 8-1GB disk drives
    16 Mbps Token Ring LAN, 2619 for Client Access
    16 Mbps Token Ring LAN, 64MB FSIOP for LAN Server/400
  - PS/2 Server 95 running NetWare 4.1 for DOS
    64MB RAM, 4 - 350 MB SCSI disk drives
    Auto LAN Streamer MC 32 Adapter
    16 Mbps Token Ring LAN

  Clients

  - ValuePoint, 486-66Mhz clients running LAN Requester from OS/2 or NetWare Requester from DOS
  - ValuePoint, 486-66Mhz clients running Client Access/400
    16 MB RAM memory
    320 KB PC cache
    CA/400 for DOS Extended (original client)
    CA/400 for OS/2 (original client)

V3R1.1 CA/400 for Windows 3.1 (premier client)
V3R1.1 CA/400 for Optimized OS/2 (premier client)

9.2 Client Access/400 File Serving Performance

With Version 3 of OS/400 the Windows 3.1* client and the Optimized for OS/2 client were introduced. Both are designed to provide integration with their respective desktop operating system, and also take full advantage of the AS/400 Integrated File System. In addition, the Windows 3.1 Client provides native Windows communication support. Only adapter drivers, such as the LAN Support Program, run as DOS TSR (terminate and stay resident) programs. Less real memory is required as a result of this change to eliminate most TSR programs.

For both the Windows 3.1 and Optimized for OS/2 clients, the Root file system offers the best file serving performance. Therefore, as users migrate to using these clients from the existing DOS clients and the original OS/2 client (16 bit), it is recommended to also move their data into the new Root file system from the QDLS file system to get the best performance. However, this is not always possible. In environments where new clients, such as Windows 3.1, and original clients, such as DOS extended, need to access the same data, the data will need to be stored in the QDLS (shared folders) file system.

Performance Data

The following performance information was collected using the **low level PC primitives** workload.

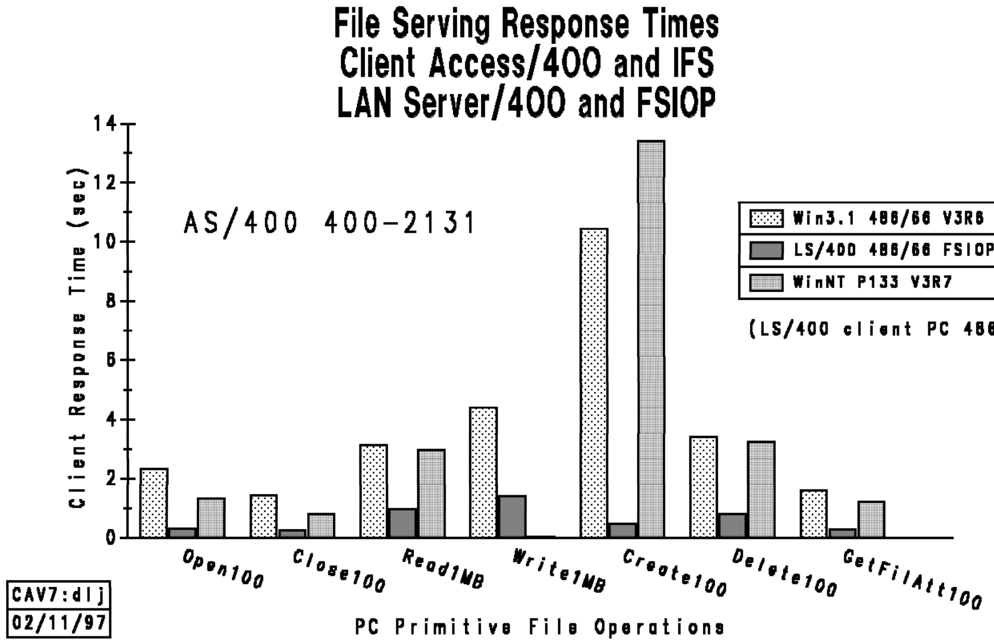


Figure 26. PC File Serving Primitives Response Time Comparison

Conclusions and Recommendations

- Performance data for Optimized for OS/2 client to Root file system is not shown but has performance characteristics comparable with Windows 3.1 to Root.
- The Root file system provides faster response times than QDLS for the Windows 3.1 client, especially for the Create100, Delete100, Open100, and Close100 test cases shown above. Similar results for Root versus QDLS can be expected for the Optimized for OS/2 client.
- Both the Windows 3.1 and Optimized for OS/2 clients provide Anynet support for running APPC based applications over TCP/IP. Using Anynet support will add to the response times shown above.
- LAN Server/400 and FSIOP provide the fastest response times compared with any of the Client Access configurations.

CA/400 for Windows 95/NT File Serving

Measurement Results:

This section provides measurement results in terms of end user response time.

Server

- AS/400 Model 50S-2121 V3R7
  256 MB Memory, 200 MB for Base Pool, 12-1GB Disk Drives
  16Mbps Token Ring LAN - 2619 IOP

Client

- Pentium 133MHz, 32 MB Memory, 1.2GB IDE Disk Drive, 16Mbps LanStreamer Token Ring Adapter
  Client Access/400 for Windows 95/NT V3R1M2
  Windows 95 with Service Pack 1 or Windows NT Workstation 4.0

Table 35. CA/400 for Windows 95/NT

CA/400 for Windows 95/NT V3R1M2
File Serving Response Times
AS/400 50S-2121 - 256MB Memory - V3R7
PC P133 - 32MB Memory

Response Time in Seconds

Primitives	**Win95 TCP/IP**	**Win95 APPC**	**Win95 IPX/SPX**	**WinNT TCP/IP**	**WinNT APPC**	**WinNT IPX/SPX**
Open	0.79	1.37	1.36	0.78	1.14	1.32
Close	0.50	1.12	1.32	0.42	0.81	1.29
Read*	20.31	35.34	177.50	15.29	21.95	187.29
Write*	17.03	24.28	99.65	10.71	23.30	222.53
GetFileAttr	0.88	1.54	1.32	1.10	2.07	2.18
Create	4.52	4.91	5.07	5.28	4.99	5.66
Delete	2.03	3.21	2.84	1.07	1.38	1.59

Note:
* - Reads and Writes are 100KB per file.
All operations are total times to execute repetitively for 100 files.
Measured in Internal IBM Lab, actual times may vary depending on configuration.

Conclusions/Explanations:

1. Windows 95/NT clients should use TCP/IP for the fastest response time. This is due to the client contribution to response time being faster with TCP/IP.
2. AS/400 server capacity is best with APPC. This is due to the AS/400 server communications pathlength difference between TCP/IP and APPC.
3. CA/400 for Windows 3.1 clients using ANYNET should experience an improvement in response time and capacity when migrating to CA/400 for Windows 95/NT using native TCP/IP. Client and server contributions to response time are generally better for TCP/IP than ANYNET for Windows 3.1.
4. Clients directly connected to the AS/400 server will generally achieve fastest response times. Gateways tend to add response time to client/server transactions. Performance-sensitive workloads usually perform best when directly-attached to the AS/400 server since the number of processing layers is minimized.
5. Clients running CA/400 V3R1M2 on Windows NT may experience better performance than clients running on Windows 95. This is due to better management of the file cache in Windows NT and other architectual differences between Windows 95 and Windows NT.
6. Section 10.4, “Tips for Improving C/S Performance” on page 184 contains additional information.

Performance Tips/Techniques for Client Access/400 File Serving

Throughout this section ′IFS File Serving′ will refer to the functions formerly known as ′Shared Folders′.

Setup and Configuration Tips &Techniques

All Clients:

1. The following should be considered when choosing the size of the cache for CA/400 clients:
   - Use a small cache (256 kilobytes - 1MB) for applications accessing data sequentially in small amounts. (or if you do not have much available PC memory)
   - Use a medium to large cache (500KB or larger) for applications accessing data randomly.
   - Use a medium to large cache (500KB or larger) for applications accessing data both randomly and sequentially.
   - The required cache size varies with the application. Creating a cache larger than is necessary may not further improve the performance of that application. However, if the caching is effectively reducing communication interaction with the AS/400 system, then the load on the AS/400 system processor is reduced.
   - If you typically run most of your applications once a day and use the data associated with those applications only once, the cache size to choose is the maximum cache size determined for those applications.
   - If you typically run some or all of your applications more than once and use the data associated with those applications more than once, the cache size to choose is the sum of the cache sizes determined for those applications.
     Note: Increased cache sizes will increase the amount of memory used on the PC. If there is already memory usage problems on the PC, increasing the cache size for CA/400 will only worsen the problem.
2. Use a large frame size if possible. Significant performance improvement may be achieved for Windows 3.1 clients when using large frame size. See Chapter 5, “Communications Performance” on page 79 for more information.

Windows 95/NT Clients:

Refer to “Performance Tips/Techniques for Client Access/400 File Serving” on page 142 for information on communications tips for Windows 95 and NT clients.

DOS Clients:

1. The following should be considered when choosing the size of the cache for DOS clients:
   - Client Access/400 supplies a tool called GETSTAT that can help you tune your PC memory cache size. The tool is in the AS/400 folder QIWSTOOL. Run the program IWSTOOL which is also in the QIWSTOOL folder and select GETSTAT to download to your fixed disk or a diskette. GETSTAT will tell you the memory used by your IFS File Serving programs and buffers. It also gives you an idea of how effectively your cache is working to limit data transfer with the AS/400 system.

Note: GETSTAT only works with the DOS Extended client.

- If using DOS Extended, use a minimum cache of 128 kilobytes.
- The value specified on the MCAC, MCAE and MCAX identifiers is used to create both the IFS File Serving cache and a cache table to keep track of things within the cache.

2. If you are using DOS Extended (XMS), space for the cache as specified on the MCAX parameter is allocated from extended memory. Using XMS does not permit the allocation of cache and the associated cache table in conventional memory.
3. If you are using DOS 5.0 and have the following statement in your CONFIG.SYS file, it is possible for DOS Extended IFS File Serving code to only use 96 bytes of your conventional memory.

```
dos=high,umb
```

4. There are two connectivity types which do support the placement of the communications buffer in extended memory: SDLC and ASYNC. In these cases the FEMU will be ignored and the buffer placed into extended memory. When possible, for performance advantage specify this in conventional memory.
5. The size of the communications buffer can have an effect on the performance of IFS file serving. The size is specified with the CBSZ identifier in the configuration file (CONFIG.PCS). Note that the size of the communications buffer can be changed only if it is located in conventional memory, with the exception as noted earlier for the 2 connectivity types that support extended memory (ASYNC & SDLC).
   - If enough memory is available on the PC, increasing the size of the communications buffer can improve response times.
   - If the PC is having problems with memory usage, use the default communications buffer size (8K).

OS/2 Clients:

1. The LAN Server/400 and the FSIOP, the IBM OS/2 LAN Server, or the PC LAN program may be better file-serving alternatives if heavy file I/O is the only function to be performed. Client Access/400 should be considered when many of the required functions are not available with other file servers. These functions include host data transfer, host integration, 5250 emulation, and remote system access.

   Since Client Access/400 can reside simultaneously with a PC file server (i.e., LAN Server/400, OS/2 LAN Server, or PC LAN Program) in the same work station, it may be desirable to install more than one server on the LAN. For example, the PC LAN Program could be used to provide the performance needed for program loading from a PC server while IFS File Serving Function could be used for storage of data for remote PCs and for data exchange between PCs and the host system.

   Another alternative is the NetWare(*) for SAA product that supports NetWare serving from a PC server. It also supports all the Client Access/400 products from an AS/400 attached to the NetWare PC server.

Application Tips & Techniques

In general, PC hard disks typically provide better performance than the IFS File Serving function. However, there are techniques to lessen performance differences. The following techniques may be used:

1. When appropriate, store some PC programs/files on the PC hard disk.
2. Copy PC files to a PC RAM disk or hard disk, use the files from the RAM disk/hard disk, and copy the changed files back to the AS/400 system when the work is complete.
3. Minimize the number of times a file is uploaded or downloaded.
4. A PC application should use an appropriate create or open operation instead of searching for the existence of a file. Searching for a file requires significant time and resources compared with other operations.
5. PC applications should be designed to open a file once, perform all necessary operations, and then close the file.
6. Use ″write with verification″ only when absolutely necessary. When verification is used, write operations are not buffered by the File Server/400.
7. When reading data from a file, read from beginning to end and not randomly. The File Server/400 buffers are used more efficiently when files are accessed sequentially from beginning to end.
8. Design your PC applications so that small files are read into memory. Memory accesses are always faster than accessing the disk or data on the AS/400 system.
9. Increase the LANMAXOUT value in the AS/400 controller description for a clients attached by a TRLAN. Changing the default value from 2 to 7 provided significant performance improvement during testing in a lab environment. Changing the default value to 6 may yield similar improvement.
10. Backing up files to a folder can take significant AS/400 resources. To avoid performance impacts to other AS/400 applications, consider the following options:

    - It is possible to use a packaging utility such as PKZIP to create a single large file which contains images of multiple files. Then the single large file can be saved for back-up purposes. This can eliminate hundreds of creations of files, and can literally make the time to back-up hundreds of times faster.
    - Back up large numbers of files to folders when system activity is low.
    - Use the CHGJOB (Change Job) command to decrease the run-time priority of the job doing the backup so that system resources are not tied up.

11. For the new clients, the Root or QOpenSys file systems offer best file serving performance.
12. Avoid doing many open/delete operations from the same Root or QOpenSys directory. Delete causes the name cache for the specified directory to be invalidated and results in poor performance. Try to keep directories that have frequent deletions (e.g. temp files) separate from directories containing files that don′t frequently get deleted.
13. Avoid any administration (create, delete, update operations) of authorization lists during peak IFS usage. This will improve the efficiency of internal caches.

14. Avoid using supplemental group profiles. Use the default value of *NONE for the SUPGPPRF parameter in the user profile. Best performance is achieved for users that do not have supplemental group profiles.
15. The Root or QOpenSys file system supports hard links and symbolic links. Hard links can improve performance by reducing the number of directory lookups. However, having many hard links to the same file can result in poor performance. Best results are achieved when you use a small number of hard links for key files that are commonly used.
16. For Optimized OS/2 clients, use ″current working directory″. This can be done by assigning a drive to the working directory or use the CD command. Using current working directory can reduce the number of directory lookups and improve performance.
17. Avoid doing many mkdir and rmvdir operations when using the Root or QOpenSys file system.
18. Avoid doing many create and delete operations when using the QDLS file system.

File Server/400 Tuning

The following tips can be used to tune File Server/400 (IFS File Server on the AS/400). Client Access/400 clients use this server when accessing network drives. There is always the risk of giving too much of any one resource to a particular job, so consider the other jobs on the AS/400 and their importance before shifting resources to the File Server/400.

As with other jobs, run priority and timeslice can be adjusted for File Server/400 jobs. For File Server/400 this can be accomplished by changing job class QPWFSERVER.

Performance of File Server/400 will also benefit from having more memory available. One option to provide File Server/400 jobs with adequate memory is to associate the QSERVER subsystem with its own pool and set aside the appropriate amount of memory for that pool.

Multiple clients executing the BAPCo workload were used to determine the optimal amount of memory needed per client. It was found that in the range of 1.5MB to 2.0MB per user, client response times ″leveled off″ such that more memory did not significantly improve response times. As the response times leveled off the paging in this pool was still higher than recommended, but since the DASD arms were at relatively low utilization the additional IO did not significantly affect response time. However, as additional memory was added to the pool beyond 2.0MB per user, the paging and faulting for that pool continued to decrease significantly until about 4.0MB was available for each client (QPWFSERV file server job to be more exact).

9.3 NetServer File Serving Performance

NetServer is a new feature in V4R2 that provides support for Windows Network Neighborhood. Using TCP/IP, NetServer provides file and print serving function for Windows 95/NT clients, Windows 3.11 clients, and OS/2 (versions 4.0 and later and file serving only) clients without any additional software on the client.

NetServer supports the Server Message Block (SMB) protocol through the use of Transmission Control Protocol/Internet Protocol (TCP/IP) on AS/400. This communication allows clients to access AS/400 shared directory paths and shared output queues. PC clients on the network utilize the file and print-sharing functions that are included in their operating systems. You can configure NetServer properties and the properties of NetServer file shares and print shares with Operations Navigator.

Clients can use NetServer support to install Client Access from the AS/400 since the clients use function that is included in their operating system.

V4R2 NetServer Performance

In V4R2 NetServer provides file serving performance comparable with client access. Work is in progress to enhance NetServer performance in a future release.

NetServer Internet Info

Additional information on NetServer is scheduled to be available at GA at the following URL: http://www.as400.ibm.com/netserver.

9.4 LAN Server/400 and FSIOP File Serving Performance

LAN Server/400 was a new AS/400 licensed program in V3R1. It works with the AS/400 File Server IOP (FSIOP) to provide high performance file serving to PC LAN clients.

LAN Server/400 implements OS/2 LAN Server technology on the AS/400 and allows the AS/400 to provide services similar to those of an OS/2 LAN Server. PC clients can access files stored on the AS/400 by using the IBM LAN Requester program, which is the same software used to request service from an OS/2 LAN Server. From the perspective of PC clients, AS/400 and OS/2 LAN Server are both servers that can store data and programs. PC LAN clients running DOS, OS/2, or Microsoft Windows are all able to request services.

LAN Server/400 and FSIOP provide high performance file serving at a level not previously possible with the AS/400 system. File serving performance with FSIOP and LAN Server/400 is comparable with best of breed PC LAN file servers. Response time and throughput characteristics of file serving with an FSIOP are consistently comparable with PC LAN file servers such as OS/2 LAN Server and Novell NetWare running on equivalent hardware. This is true for a small or large number of users.

The remainder of this section gives performance information for several scenarios using LAN Server/400 and FSIOP. For a thorough understanding of LAN Server/400 features and function, the following IBM ″redbook″ is recommended: ***LAN Server/400 - A Guide to Using the AS/400 as a File Server***. The document number is GG24-4378. Another recommended document is ***LAN Server for OS/400 Administration SC41-3423***.

QLANSrv is a file system which will be present on systems that have the LAN Server/400 licensed program installed. QLANSrv provides a mechanism for host applications and the OS/400 Integrated File System (IFS) file server (File Server/400) to access data which is in the LAN Server/400 file system. **Host applications or PC clients (using File Server/400) which access files through QLANSrv will not achieve the same level of performance that PC clients using LAN Requester applications can achieve.**

LAN Server/400 and FSIOP provide equivalent performance with a 2617 or 2619 IOP when used as a LAN communications controller. The FSIOP has greater capacity for large transfer scenarios - achieving rates over 14 Mbps. Please refer to 5.2, "LAN Protocols, Lines, and IOPs" on page 82 for additional information comparing FSIOP (6506 IOP) with 2617 and 2619 IOPs.

LAN Server/400 and FSIOP Sizing Guidelines

Estimating the Number of Clients supported by an FSIOP

The number of clients that an FSIOP can support when used as a file server depends on the amount of memory available for the HPFS cache on the FSIOP, the rate of requests from the clients, client hardware configurations, LAN utilization, and other factors. It is difficult to generalize with so many variables.

The following estimates for the number of users supported are only guidelines. For casual file serving, the number of supported users will be larger. For very heavy file serving (e.g. multimedia continuous medium-quality video which delivers data to each client at a rate of 150Kbytes per second), the number of supported users will be lower, as resources such as the LAN become a bottleneck.

Table 36. AS/400 FSIOP - Getting Started

Guidelines for Getting Started	
FSIOP Memory Size	**Number of Users**
16 MB	1 - 20
32 MB	20 - 50
48 MB	50 - 100
64 MB	100 - 250

AS/400 CPU Requirements

The FSIOP is supported by all AS/400 Advanced Series and AS/400e series models. Because relatively little AS/400 system CPU is consumed on behalf of file serving requests generated by LAN Requester clients, the various AS/400 models will typically yield similar file serving performance characteristics. LAN Server/400 will provide more file serving throughput at significantly less CPU as compared with Client Access file serving.

To minimize the use of AS/400 CPU resource when doing read operations from FSIOP-attached clients, large file system caches should be used on the FSIOP. The section below on DASD I/O Requirements discusses HPFS cache sizes for the FSIOP. The data in table Table 37 on page 149 below shows the effect on AS/400 CPU and disk resources when running a fixed workload using FSIOPs of three different memory sizes.

Table 37. Effects of FSIOP memory size on AS/400 resources for a fixed workload

LAN Server/400-FSIOP AS/400 400-2131 V3R6			
FSIOP Configuration	**AS/400 CPU Utilization**	**AS/400 I/Os per second**	**BAPCo5 Average Completion Time**
16MB	8.0	19.5	719
32MB	4.5	10.9	691
64MB	3.3	7.5	680

Note:
Average utilization of 16Mbps LAN was 9% for all tests
Average utilization of FSIOP ranged from 28% to 32%

AS/400 System DASD I/O Requirements

To minimize the impact on the AS/400 disk I/O′s it is best to make maximum use of the HPFS cache on the FSIOP by ensuring that sufficient FSIOP memory is available for HPFS cache. The data in Table 37 shows fewer disk I/Os happening with larger FSIOP memory, since the HPFS cache is larger. HPFS cache is used by the High Performance File System for data being read from and written to disk. In the IBM Redbook *LAN Server/400 - A Guide to using the AS/400 as a File Server (GG24-4378)* a detailed description is given on how the memory on the FSIOP is allocated. HPFS memory allocations for various FSIOP configurations are shown below:

Table 38. AS/400 FSIOP HPFS Cache Sizes

FSIOP Memory Size (MB)	**HPFS Cache Size (MB)**	
	1 Port	**2 Ports**
16	4.8	2.8
32	16.3	13.8
48	28.6	25.6
64	43.6	40.6

Notice that for the 16MB FSIOP with two ports, less than 3MB is available for HPFS cache.

HPFS Cache statistics and other performance counters are available from the FSIOP. This data can be collected using the STRPFRMON command. ***IBM AS/400 Work Management SC41-3306*** provides information on the data collected by the STRPFRMON command.

AS/400 System Memory Requirements (machine pool)

Using an FSIOP with one port will require approximately 2700k bytes from the AS/400 system pool #1 (machine pool) exclusively for FSIOP and LAN Server/400. Using an FSIOP with two ports will require approximately 4200k bytes from system pool #1 exclusively for FSIOP and LAN Server/400. These memory requirements are different from the memory on the FSIOP card itself. Please refer to Chapter 12, “OS/400 Integration of Lotus Notes Performance” on page 201 for updated memory guidelines for Integrated PC Servers. The information is included under a section labeled ″AS/400 Machine Pool Guidelines″.

BAPCo5 Workload File Serving Comparisons

The data for the following chart was collected using the BAPCo5 workload which is described at the beginning of this chapter. Because the clients run with zero think-time scripts, the load generated on the AS/400 server by these clients is representative of a load which would typically be generated by a larger number of clients.

The following chart shows the relative positioning of file serving performance using the following client --> server configurations:

- Client Access Optimized for OS/2 client --> IFS root file system
- OS/2 LAN Requester 4.0 --> LAN Server/400 and FSIOP
- NetWare requester 4.1 --> NetWare 4.1 on PC Server95 running DOS

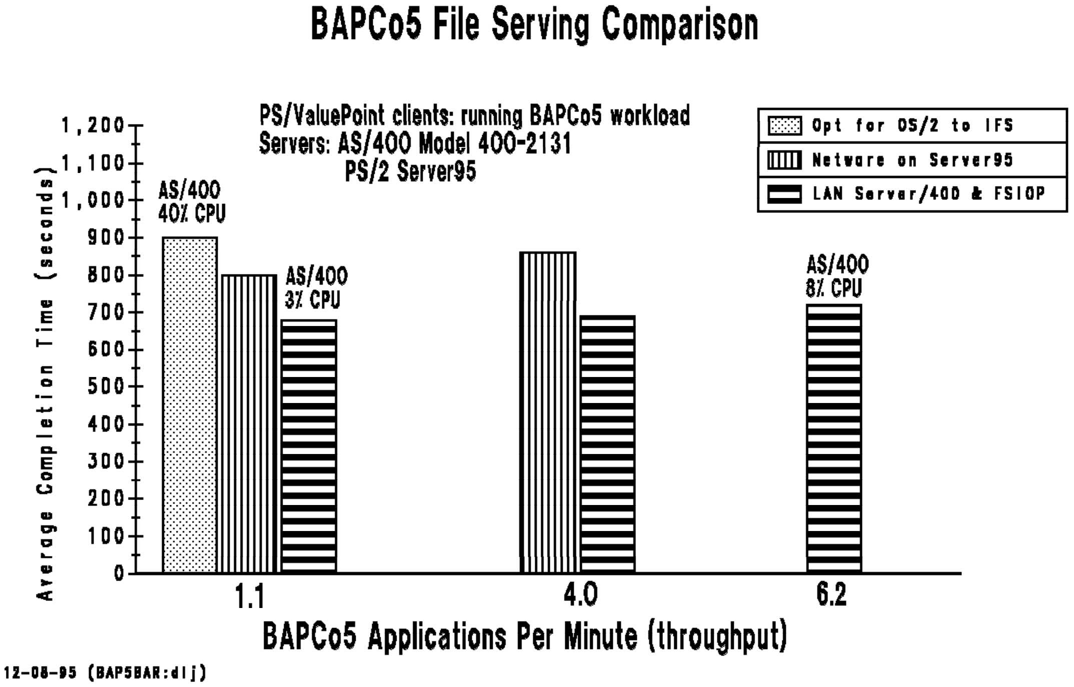


Figure 27. PC File Serving Using 5 PC Applications

Conclusions and Recommendations

- LAN Server/400 and FSIOP provide high performance file serving competitive with PC Network servers such as NetWare. Data was estimated for LAN Server/400 running at a rate of 1.1 BAPCo5 applications per minutes for purposes of comparing with client access running to the Integrated File System. While end user response times are fairly close at this point, the AS/400 CPU utilization is much lower for LAN Server/400 (3% versus 40%), since the AS/400 CPU is primarily used when the FSIOP accesses files on AS/400 DASD. A wide range of AS/400 CPU models will yield similar response time results for the FSIOP workload. This is because the CPU and disk resources on the AS/400 system are used in a fashion that is generally asynchronous to individual client requests. Of course the overall capacity (how many FSIOPs can be supported) will vary based on the AS/400 CPU.
- Data for NetWare is provided as a comparison. For four of the five applications which are part of the BAPCo5 workloads, the response times were very similar for LAN Server/400 and NetWare. The fifth application was limited (by NetWare and/or the NetWare requester) to sending very small frames of information. This contributed to the overall response times being slightly longer for the NetWare configuration than for the LAN Server/400 configuration.
- It is estimated that at 4 BAPCo5 applications per minute on the chart, a load approximately equivalent to 100 file serving clients is being generated. For more casual file serving clients, this rate of throughput could represent as many as 200 clients.
- Near 4 BAPCo5 applications per minute, the FSIOP utilization was 46%. This utilization is shown in the AS/400 Performance Tools LPP component report. Other FSIOP performance data collected by STRPFRMON can be queried in the file QAPMIOPD. Other than IOP utilization, data from the QAPMIOPD file is not available through any of the Performance Tools reports and must be queried. The file QAPMIOPD contains a large number of statistics associated with the FSIOP, including the software operating within it - LAN Server/400 and the High Performance File System, OS/2 utilizations, and FSIOP CPU utilizations. For more information on QAPMIOPD file field descriptions, please refer to Appendix A of the Version 3 Work Management Guide (SC41-3306-00).
- Client Configuration

  The speed and configuration of LAN Requester clients can have a big impact on file serving capacity. When a larger share of the processing can be done on the client, the more perceived capacity the file server has. Faster clients are generally able to generate a higher volume of file serving requests, depending on the PC application being used.

  For the BAPCo5 workload, part of the average completion time is spent on the client. Therefore, the configuration of the client (e.g. processor speed, amount of memory, etc.) affects the results. A slower client or one with less memory will have longer completion times than shown in this chart. These measurements were performed using PS/ValuePoint 486-66Mhz clients. Each PS/ValuePoint client was executing the BAPCo5 workload with zero think-times which has the effect of simulating many PCs per client.

  Based on tests using a subset of the BAPCo5 workload, OS/2 LAN Requester provided slightly faster response times than the DOS LAN Requester.

The following table provides additional information collected for the BAPCo5 file serving comparisons.

Table 39. File Serving Performance using 5 PC Applications

Client Access/400, LAN Server/400-FSIOP, NetWare
Servers: AS/400 400-2131 or PS/2 Server95

Environment	BAPCo5 Apps Per Minute	AS/400 CPU Util (%)	16Mbps LAN Util (%)	FSIOP Util (%)	BAPCo5 Avg Completion Time (sec)
CA/400 Opt OS/2	1.1	40	03	--	910
LAN Server/400	1.1(1)	03(1)	03(1)	18(1)	690(1)
LAN Server/400	4.0	05	17	46	710
LAN Server/400	6.2	08	26(2)	64	730

Note:
(1) this data estimated for comparison with CA/400 Opt OS/2
(2) data not available and is estimated

9.5 Multimedia File Serving

This section will provide performance information for multimedia file serving using V3R6 and Ultimedia System Facilities (USF).

USF is a feature of OS/400 that brings multimedia function to the AS/400. Image, audio, and motion video can easily be added to new or existing applications executing in either the client or AS/400 Host/Server using the provided APIs. Repository and object services support provides the capability to capture, register, and play back multimedia objects such as video clips, audio files, and still images. A repository is provided to catalog and track objects. Additional capabilities to edit, transform and sequence objects give a complete support platform on which to build multimedia enabled applications. USF also extends the supersedes the capabilities covered by the Ultimedia Video Delivery System/400.

The USF product requires the use of a PC client running either OS/2 2.1 with Multimedia Presentations Manager/2, or DOS 5.0 and MS Windows 3.1 with Multimedia Extensions. AS/400 host based applications or PC based C language applications can make use of the provided API′s. Client Access/400 is required and additional multimedia hardware and software may also be required depending upon the type of multimedia data to be used.

The following workloads were used to evaluate multimedia file serving performance.

Workload Descriptions

1. Heavy Image
   a USF sequence object displays a bitmap image every 15 seconds (150KB average size)
2. Heavy Image with Audio
   a USF sequence object displays a bitmap image every 15 seconds, an audio clip (16 bit 22Khz) plays continuously in the background while the images (150KB average size) are displaying
3. Video (high) - Continuous High Quality DVI Video
   a sequence object repeatedly invokes a production level DVI video file which requires 180KB/sec average transfer rate (this constitutes a high quality DVI file), files are varied to ensure file accesses are from disk
4. Video (medium) - Continuous Medium Quality DVI Video
   a sequence object repeatedly invokes a DVI video file which requires 150KB/sec average transfer rate (this constitutes a medium quality DVI file), files are varied to ensure file accesses are from disk

The following configurations were used for the multimedia tests described in this section.

Configurations

- Server: 9406 Model 510-2143 running V3R6

- 256MB memory, 2-6606 (1967MB) and 1-6602 (1031MB) DASD
- 2619 Token Ring LAN IOP, used at 16Mbps
- frame size = 16393, RU size = 16384

- Client: PS/2 Model 95 - 50Mhz
  - tested with both WARP and Windows 3.1
  - 32MB memory
  - 4/16 Mbps Token Ring adapter, used at 16Mbps
  - Action Media II adapter

Measurement Results

The Optimized for OS/2 Client Access/400 client was used to collect the following information. Measurement data shown below was reported by the OS/400 performance monitor.

Table 40. V3R6 Multimedia File Serving Performance Data

Ultimedia System Facilities
Server: AS/400 510-2143 V3R6
Client: Optimized for OS/2 CA/400 running USF

Scenario	AS/400 CPU Util (%)	AS/400 Disk IOP Util (%)	AS/400 Disk Util (%)	AS/400 TRLAN 2619 IOP Util (%)	16Mbps TRLAN Util (%)
Heavy Image	1.0	0.9	1.7	4.9	< 1
Heavy Image with Audio	2.3	1.7	2.7	7.2	2.0
Video (high)	2.3	3.3	5.4	8.4	8.0
Video (medium)	2.1	3.6	4.9	7.8	7.0

Note:
the utilizations above were generated using a single client

Conclusions and Recommendations

- The results from Table 40 give an indication of how AS/400 system resources are used for a single client executing the various scenarios described.
- Performance when using the CA/400 for Windows 3.1 client was not as good for these and other scenarios. Video playback was noticeably worse than when using the OS/2 client, and may not be acceptable for software motion video using a similar client configuration. Using the Action Media II adapter improved the video quality. This adapter may no longer be available, but other digital video cards can be used to provide hardware assist for off-loading decompression.
- The USF product is built on the QDLS file system and uses QDLS to store object and attribute information for all USF objects. The Integrated File System introduced in V3R1 provides improved performance to the new Client

Access/400 clients, and USF allows for files to be served from the Root file system as well as QDLS. Performance improvements can be realized by serving video files from the Root file system. Please refer to the section on **Client Access/400 File Serving Performance** earlier in this chapter for additional information.

- Using the largest frame and RU sizes possible is a necessity when running multimedia applications. 16K frames and RU sizes were used for all measurements. The following steps can be used to adjust the settings for the indicated client.
  - **Optimized for OS/2 client:** click on AS/400 icon on desktop, click on CA/400 Connections icon, select AS/400 server, click on SNA Configuration, click on LAN, click on Advanced Options, set maximum I field size to 16393, click on Mode, select QPCSUPP and click on change, click on Max RU size and enter 16384, click on OK, close and save settings.
  - **Windows 3.1 client:** click on CA configuration, click on System, select AS/400 server, click on Global Options, set maximum frame size to 16393.
- For client hardware, ensure that reference configuration (for microchannel) or switch settings (for non-microchannel) specifies at least 16KB RAM for the token ring card. Older cards sometimes come configured for 8KB, which will limit data throughput and can cause jerky video.

AS/400 Performance Capabilities Reference (V3R2), ZC41-8166 contains additional performance information comparing Client Access and LAN Server/400 multimedia file serving performance.

9.6 QLANSrv Save/Restore Performance for V4R1

For V4R1, QLANSrv Save/Restore performance has been improved by up to 300% over previous releases. Actual results will depend on your environment.

Save/Restore Rates for QLANSrv Storage Spaces

The save/restore rates quoted below are expressed in terms of megabytes per hour (MB/Hr). The measurements were done on an AS/400 model 400-2131 CPU using either a 6506 FSIOP (66MHz 64MB) or a 6616 FSIOP (166MHz 256MB). On the AS/400 the SAV commands were run in non-restricted state in the *Interact pool (50 MB and activity level 3). The 6380 measurements were done with USEOPTBLK=*NO, DATCPR=*YES and COMPACT=*NO. The 3590 measurements were done with USEOPTBLK=*YES, DATCPR=*DEV and COMPACT=*DEV. These parameters were selected to provide optimal performance rates.

The following QLANSrv storage spaces were used to obtain the save/restore rates:

Table 41. QLANSrv Storage Space Definitions

Name	Size (MB)	Objects	Description
APPS	94	2383	Applications
USERS	48	3883	User Profiles
A1GB	1000	14370	1 GB Space
A500MB	500	1	500 MB object

Table 42. Save/Restore rates for QLANSrv with 6506 FSIOP - V3R7

Storage space	Tape Device	Save (MB/Hr)	Restore (MB/Hr)
APPS	6380	297	267e
	3590	322	290e
	*SAVF	333	303
USERS	6380	123	111e
	3590	136	122e
	*SAVF	138	125
A1GB	6380	398	358e
	3590	458	412e
	*SAVF	477	431
A500MB	6380	465	419e
	3590	755	680e
	*SAVF	727	656

Note: e - estimated

Table 43. Save/Restore rates for QLANSrv with 6506 FSIOP - V4R1

Storage space	**Tape Device**	**Save (MB/Hr)**	**Restore (MB/Hr)**
APPS	6380	666	599e
	3590	720	648e
	*SAVF	740	666
USERS	6380	271	245e
	3590	274	247e
	*SAVF	277	250
A1GB	6380	796	716e
	3590	1070	963e
	*SAVF	1064	959
A500MB	6380	973	878e
	3590	2270	2043e
	*SAVF	2064	1858
Note: e - estimated			

As can be seen from the previous charts, V4R1 QLANSrv changes provide up to a 300% improvement in save/restore performance over V3R7. The improvement ranges from 200% for saving USERS on the 6380 tape to 300% for saving A500MB on the 3590 tape.

Table 44. Save/Restore rates for QLANSrv with 6616 FSIOP - V4R1

Storage space	**Tape Device**	**Save (MB/Hr)**	**Restore (MB/Hr)**
APPS	6380	852	767e
	3590	1029	926e
	*SAVF	1068	961
USERS	6380	400	360e
	3590	410	369e
	*SAVF	421	380
A1GB	6380	903	813e
	3590	1390	1251e
	*SAVF	1388	1251
A500MB	6380	977	879e
	3590	2387	2148e
	*SAVF	2284	2056
Note: e - estimated			

The faster 6616 FSIOP provides up to 40% better save/restore rates than does the 6506 FSIOP for QLANSrv save/restore.

The 2850 FSIOA (133MHz 128MB) provides similar QLANSrv save/restore rates as the 6616 FSIOP.

Memory Tuning for QLANSrv Save/Restore

In terms of memory requirements, QLANSrv save/restore are sensitive to the size of the memory pool where it is executing. To achieve acceptable save performance, the following two calculations must have a result equal to or greater than 15 MB:

- The pool size divided by the activity level of the pool.
- The pool size divided by 2.

For example, if the pool size is 30 MB and the activity level is 5, then the above two calculations would be 6 MB (pool size divided by the activity level) and 15 MB (pool size divided by 2). In this case, you would want to either increase the pool size to 75 MB or change the activity level to 2. You should refer to the Work Management Guide for information on changing pool sizes and activity levels.

9.7 WarpServer/400 File Serving Performance

The main objective here is provide information which confirms performance parity (or better) to similarly configured systems:

- IPCS with WarpServer/400
- Standalone PC with WarpServer

(Please see the introductory remarks in 9.1, " File Serving Performance Positioning" on page 137) With this product, we wanted to provide performance metrics which:

- users of PC file servers could easily identify with and use
- used simpler workload scenarios.
- used workloads that were repeatable.

To that end, we discontinued the use of the BAPCo5 workload (see section "File Serving Workloads and Configurations" on page 138). We will be presenting data from the following three(3) workloads/environments:

1. NetBench(R) 5.0, a portable, publicly available benchmark by Ziff-Davis Publishing Company.

   ″NetBench″ is a registered trademark of Ziff-Davis Publishing Company.

2. Large file transfer (download of a 264MB file)
3. Low Level PC Primitives

   (see ″Primitives″ discussion in "File Serving Workloads and Configurations" on page 138)

Configurations of Servers & Clients

This section will describe all the equipment used in the three environments listed above. Names will be assigned to allow easy cross referencing. The following acronyms and codes will be used:

- IPCS (aka FSIOP) - Integrated PC Server. We will use the type codes for each:
  - 6616 is an IPCS with a Pentium 166mHz processor. Memory sizes can extend up to 256MB.
  - 6506 is an IPCS with an Intel 486DX2 66mHz processor. Maximum memory size is 64MB
  - 2850 is an IPCS that is available only with RISC (AS/400 Model 150) systems. It has Pentium 133mHz processor and maximum memory size is 128MB.
    - The model 150-2269 was configured with 96MB memory; DASD=4 units of 1.9GB; V4R1 OS/400.
- RISC refers to the AS/400 processor type. V3R6 and later systems are of this type.
  - The model 400-2131 was configured with 96MB memory; DASD=12 units of 1GB; V4R1 OS/400.
- IMPI refers to the original AS/400 processor design. V3R2 and earlier systems are of this type.

- The IMPI model E25 was configured with 64MB memory; DASD=8 units of 1GB + 5 units of 320MB; V3R2 OS/400.

- PC/166/RAID - an IBM PC 750 (M/T 6887-86B) was configured with Pentium 166mHz; 64MB memory; DASD= 1.2GB; NIC = TRN 16/4 Lanstreamer PCI; Attached SCSI RAID unit - IBM 3517001; 5 units of 1GB DASD; Level 0;
- PC/133/RAID - similar to the preceding except for Pentium 133mHz.
- SUT - Server (or system) Under Test. Each of the above may be configured to be a SUT by plugging it into the Network - in our case we use TRN (Token Ring Network) rated for 16 mbps(megabits per second). The SUT must be loaded with the appropriate server software - either WarpServer/400 (WS) or LANServer/400 (LS). PC servers are loaded with OS/2 WarpServer 4.

  The PC/166/RAID and PC/133/RAID systems are ′desktop′ PCs extended with the SCSI RAID box of 5x1GB DASD arms. These are probably not the ′ultimate′ driving machine in terms of professional PC Servers - but availability rules!
- NIC - Network Interface Card(s) used in SUTs are TRN 16 mbps PCI LanStreamer.
- Client PCs - these are IBM PC 350s; Pentium 133mHz; 32MB; M/T 6586-79H; DASD= 1.2GB; NIC=Auto 16/4 ISA
- NetBench Controller(s) Valuepoints; i486DX2 66mHz; 32MB; M/T= 6384-v73; DASD=340MB; NIC= Auto 16/4 ISA
- Client software = Windows95 with IBM NetWork Client for Warp/LAN Server connectivity;

NetBench Workload

Description This test environment consists of a TRN with a SUT, 10 PC Clients and a Controller PC with NetBench(R) version 5.0 installed.

Brief Look at NetBench: NetBench(R) is a test environment developed and licensed from Ziff-Davis. It drives a ′file serving′ workload from a set of Client PCs to a file server such as the AS/400 IPCS with WarpServer/400. This workload mimics the network file operations of the following Windows client applications and reports NetBench scores (MB/second):

- Adobe(**) PageMaker(**) 6.0
- Borland(**) Paradox(**) 7.0
- CorelDRAW!(**m) 6.0
- Lotus(**) Word Pro(**) 96
- Microsoft Access (**) 7.0
- Microsoft Excel (**) 7.0
- Microsoft(**) PowerPoint(**) 7.0
- Microsoft Word (**) 7.0
- Windows 95 Explorer (**)

The standard test suite NBDM_60.TST provided in NetBench was used in our tests. More information is available on the Ziff-Davis Benchmark Operations′ Web site: ***www.zdnet.com/zdbop***

The following SUT(Server Under Test) configurations were used:

1. ***SUT1*** - 6616/256MB; RISC(400-2131); WarpServer/400
2. ***SUT2*** - 2850/128MB; RISC(150-2269); WarpServer/400
3. ***SUT3*** - 6506/64 MB; RISC(400-2131); WarpServer/400
4. ***SUT4*** - 6506/64 MB; CISC(Model E25); LANServer/400
5. ***SUT5*** - PC/Pentium-166; 64 MB; RAID; WarpServer
6. ***SUT6*** - PC/Pentium-133; 64 MB; RAID; WarpServer

Ten client PCs were used to drive each of the above SUTs using the NetBench workload/environment.

- ***CLPC1*** - PC350/133mHz; 32MB; 1.2GB DASD; 16/4 ISA; Windows95; IBM NetWork Client for Warp/LAN Server;

Results The results of NetBench measurements are presented as ratios of SUT scores - i.e. allowing the various SUTs to be compared one to another. The purpose of this tabulation to convey the ′general′ overall sense and confirmation that IPCSs do perform as well as equivalently configured PC-based servers in an interactive file serving environment.

Table 45. Performance Thruput Ratios between Servers(SUTs)

Performance Ratios between WARP Servers
SUT Configurations defined above.

Performance Ratios between WarpServer Platforms

----	SUT1	SUT2	SUT3	SUT4	SUT5	SUT6
SUT1	1.00	0.96	1.39	1.23	1.21	1.28
SUT2	1.05	1.00	1.45	1.28	1.26	1.34
SUT3	0.72	0.69	1.00	0.88	0.87	0.92
SUT4	0.82	0.78	1.13	1.00	0.98	1.05
SUT5	0.98	0.79	1.15	1.02	1.00	1.06
SUT6	0.78	0.74	1.08	0.95	0.94	1.00

Note:

- Ratios are Row values divided by Column values; e.g. the value 1.13 found in the intersection of row 4 (SUT4), column 3 (SUT3)
  - by dividing the throughput values of SUT4 by the throughput value of SUT3
  - e.g. SUT4′s throughput is 1.13x (or 13% more) than SUT3.

  Measured in Internal IBM Lab, actual values may vary for individual configurations.

Conclusions: The main ′comparables′ are the:

1. Pentium IPCS(SUT1, SUT2) versus the Pentium PCs(SUT5, SUT6): The IPCS has up to a 30%+ advantage - however ′your mileage may vary′. The AS/400 advantage will be:
   - better DASD handling (multiple/parallel DASD tasks)
   - more DASD arms than generally found on a PC server.
   - more DASD caching - a memory statement
   - some offloading of processor load to AS/400 (DASD handling)
2. differences of 5% should be attributed to measurement variations.
3. SUT1, SUT2 utilizations and additional information:
   - Pentium and AS/400 RISC CPU% at about 30%

- Experiments with reduced IPCS memory (64MB) did not change the throughput; indicating that this particular configuration of NetBench did not present a load requiring more than 64MB memory.
- TRN bandwidth was highly utilized at about 77%. Experiments with two TRN rings increased the throughput scores such that the ratios remained about the same.
- DASD utilization at about 20%

Large File Download Results

Description This test environment consists of a TRN with a SUT, and a Client PC. The workload consists of downloading a large file from the SUT to the Client. The file is a flat (text) 264 MB in size.

The goal here is also for the IPCS configurations to perform as well as a standalone PC.

These are the SUT configurations tested for comparative performance:

1. ***SUT1a*** - 6616/ 64MB; RISC(Model 2131); WS
2. ***SUT1b*** - 6616/128MB; RISC(Model 2131); WS
3. ***SUT1c*** - 6616/256MB; RISC(Model 2131); WS
4. ***SUT2a*** - 2850/ 64MB; RISC(Model 150-2269); WS
5. ***SUT2b*** - 2850/128MB; RISC(Model 150-2269); WS
6. ***SUTpc*** - PC/166 64 MB; RAID; WS

The following is the client PC used to drive each of the above SUTs:

- ***CLPC1*** - PC350/133mHz; 32MB; 1.2GB DASD; 16/4 ISA; Windows95; IBM NetWork Client for Warp/LAN Server;

Each of the above configurations achieved similar throughput rates:

- 1800 +/- 50 megabytes/hour
- Utilization of Pentium (6616) at below 5%
- Utilization of AS/400-2131 at about 20%
- Utilization of LAN at about 25%

Conclusions: In this environment, each of the above configurations provided very similar throughput rates; in particular, the IPCS performed comparably to the standalone PC.

Low Level PC File Primitives Results

Description This test environment consists of a TRN with a SUT, and a Client PC. The workload consists of running the program IOTLL which performs a series of low level file accesses from the Client PC to the SUT. Timings for the various operations are presented below. (see ″Primitives″ discussion in “File Serving Workloads and Configurations” on page 138)

The goal here is also for the IPCS configurations to perform as well as a standalone PC.

These are the SUT configurations that were tested for comparative performance:

1. ***SUT1c*** - 6616/256MB; RISC(Model 2131); WS
2. ***SUT2b*** - 2850/128MB; RISC/(Model 150-2269); WS
3. ***SUT3a*** - 6506/64MB; RISC/(Model 2131); LS
4. ***SUT5a*** - PC/166 32MB; RAID; WS
5. ***SUT5c*** - PC/166 128MB; RAID; WS

The following is the client PC that was used to drive each of the above SUTs:

- ***CLPC1*** - PC350/133mHz; 32MB; 1.2GB DASD; 16/4 ISA; Windows95; IBM NetWork Client for Warp/LAN Server;

Table 46. File Primitives

File Primitives on WARP Server- Comparisons
Response times of File Operations to
the PC type servers
SUT Configurations defined above.

Response Time in Seconds

Primitives	**SUT1c**	**SUT2b**	**SUT3a**	**SUT5a**	**SUT5c**
Open	0.22	0.22	0.27	0.17	0.16
Close	0.16	0.16	0.22	0.22	0.22
Read*	10.05	9.61	10.44	10.88	10.93
Write*	20.38	20.32	21.86	23.94	22.74
GetFileAttr	0.17	0.16	0.22	0.17	0.22
Create	0.22	0.17	0.33	0.33	0.22
Delete	0.17	0.16	0.28	0.17	0.22

Note:
All operations are total times to execute repetitively for 100 files.
Measurement variability is approximately +/- 15%.
Measured in Internal IBM Lab, actual times may vary depending on configuration.

Conclusions: The above table indicates that all the SUTs basically performed equivalently - in particular, the IPCS with WarpServer

Chapter 10. DB2/400 Client/Server and Remote Access Performance

With the announcement of the AS/400 Advanced Servers using PowerPC technology, IBM has again provided a clear direction for customers that are moving to a client/server environment. With V4R1, overall system performance of the high-end AS/400 server model S40 12-way increased significantly over the existing high-end Model 53S, giving outstanding growth and improved price/performance. For customers who have a mixed environment (a combination of fixed function workstations and PC′s), then AS/400 Advanced System models using PowerPC technology provide significant system performance growth and improved price/performance (see Chapter 2, "AS/400 System Capacities and CPW" on page 3 for relative performance of these models).

When using client/server technology, it is important to consider the impact of the various client and server components, and their effect on performance. There are different ways of implementing client/server applications. In this chapter, guidelines are provided for a number of common implementation strategies, to help understand the impact of the performance of the client system and the server system using database serving workloads.

With the introduction of Java on the AS/400, database access and client/server development have changed to an even more open environment. The AS/400 Toolbox for Java provides a JDBC driver that conforms to the JDBC specification published by Sun Microsystems. JDBC enables application developers to write portable applets and applications that access Relational Database information.

Client Access/400 contains an OLE/DB driver for the AS/400 in V4R1. This driver allows developers to easily and quickly develop client/server applications for the AS/400.

ODBC will continue to be strongly supported by IBM as an ***open*** way of connecting to DB2/400.

In general, V4R2 database serving performance will be equivalent to V3R2 when comparing AS/400 models with the same CPW value. The new models using PowerPC technology in this case provide significant system performance growth and improved price/performance. See "Server Challenge Benchmark (SCB)" on page 173 for more information on AS/400 release comparisons.

Only V4R1 and V4R2 performance results are included in this document. Use the information provided in *AS/400 Performance Capabilities Reference (V3R2), ZC41-8166*, Chapter 7, ″DB2/400 Client/Server and Remote Access Performance Information″, as a guide for V3R2 performance. In addition, refer to "Related Publications / Documents" on page vii on how to access a presentation that covers *AS/400 Versus Microsoft′s SNA Server Gateway*.

10.1 Client Performance Comparisons

Under different client server implementations, PC hardware configuration plays an important role in overall performance. Figure 28 shows the relative client performance for various PC configurations. This figure also shows the client contribution to response time for different client/server implementations.

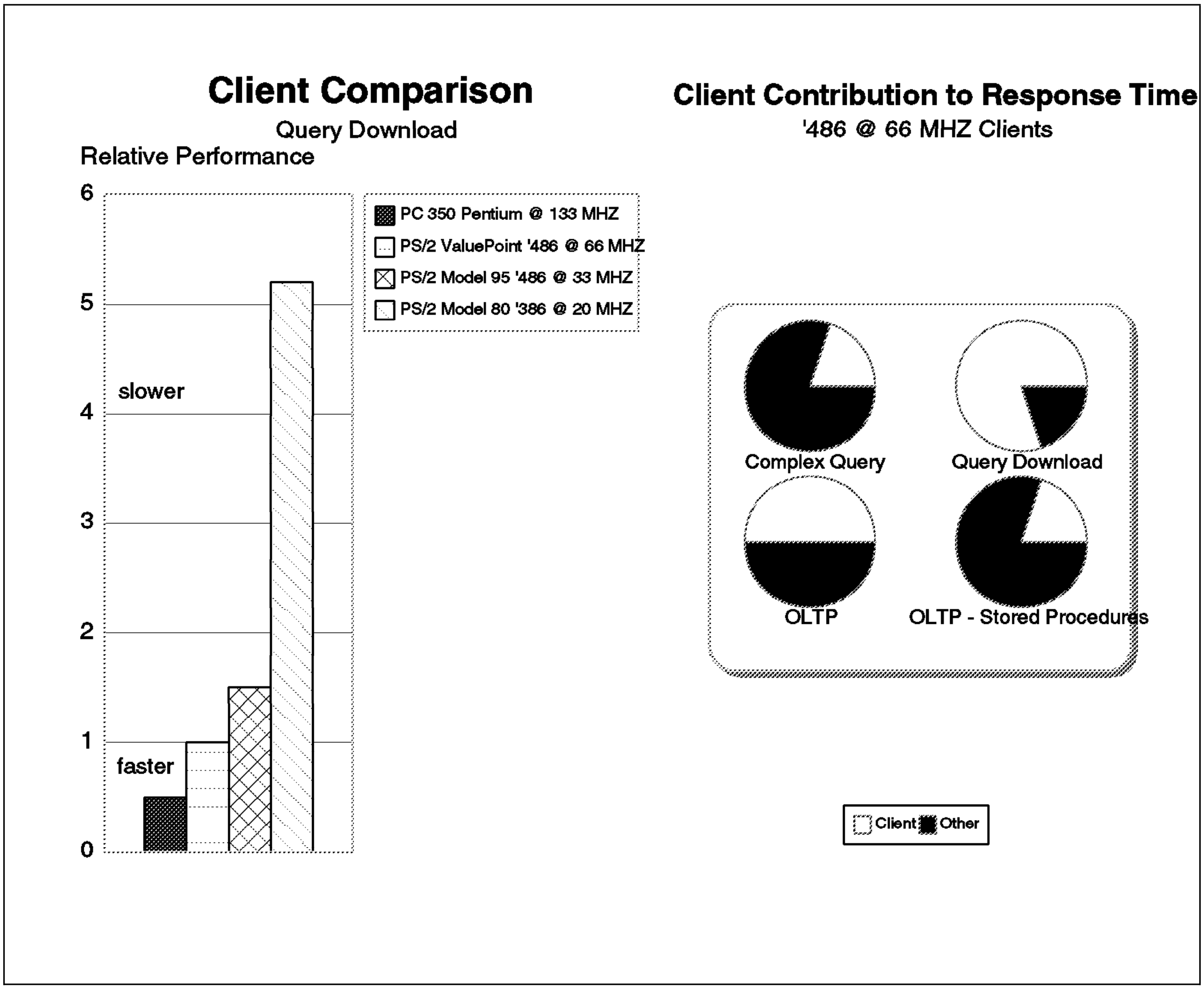


Figure 28. Client Performance Comparison

Figure 28 shows the effect of the client on overall response time. The chart to the left shows the performance of three different PC configurations, when performing a query download. In general, a faster PC CPU will improve performance, but there are other issues which should be taken into account such as disk drive performance, main storage, memory cache etc. If an acceptable relative performance value was 2.0 (or 2X slower than a ′486 @ 66 MHZ), then the PS/2 model 80 would prevent this environment from achieving the required response time criteria.

The chart to the right shows the client contribution to response time for the following client/server scenarios:

- A complex query
- A query download

- OLTP
- OLTP using stored procedures.

For applications where 50% or more of the application response time contribution is on the PC client, such as a query download or an OLTP application, better performance can be achieved by focusing on the client performance and selecting a faster 486, Pentium (**), Pentium Pro (**), or Pentium II (**) processor.

It is also important to note that client memory can be a significant component of response time as well. For the Windows 3.1 client, most database serving operations will perform acceptably if the client has at least 8 MB of memory. The Windows 95 client performs acceptably with 16MB of memory, and the Windows NT client performs acceptably with 32MB of memory. The difference in memory is due to the overhead associated with the different operating systems. Client/Server operations that operate on a large amount of data will usually perform better if the client has more than the amount of memory suggested above.

When most of the response time contribution is on the AS/400 such as a complex query, or when processing an OLTP stored procedure, greater performance improvements may be achieved by optimizing the AS/400 application or upgrading AS/400 hardware. For example, it may be possible to create a logical view for a query which is frequently executed.

10.2 AS/400 Toolbox for Java

The AS/400 Toolbox for Java is a set of enablers that supports an internet programming model. It provides familiar client/server programming interfaces for use by Java applets and applications. The toolbox does not require additional client support over and above what is provided by the Java Virtual Machine and JDK.

The toolbox provides support similar to functions available when using the Client Access/400 APIs. It uses sockets connections to the existing OS/400 servers as the access mechanism for the AS/400 system. Each server runs in a separate job on the AS/400 system and sends and receives architected data streams on a socket connection.

The AS/400 Toolbox for Java is delivered as a Java package that works with existing servers to provide an internet-enabled interface to access and update AS/400 data and resources.

The base API package contains a set of Java classes that represent AS/400 data and resources. The classes do not have an end-user interface but simply move data back and forth between the client program and an AS/400 system, under the control of the client Java program.

For more information on the AS/400 Toolbox for Java see the Redbook *Accessing the AS/400 System with Java* Document Number SG24-2152-00.

JDBC Driver

The JDBC driver that is included in the AS/400 Toolbox for Java allows database access to the AS/400 using APIs that are similar to ODBC. This JDBC driver talks to the same server job on the AS/400 as the ODBC driver included with Client Access/400. Many of the options for the Client Access/400 ODBC driver are therefore included in the JDBC driver. Also, any of the database and communication tuning for ODBC and Client Access/400 can be used for JDBC and the Toolbox.

JDBC allows SQL statements to be sent to the AS/400 system for execution. If an SQL statement is run more than one time, use a PreparedStatement object to execute the statement. A PreparedStatement compiles the SQL once, so that subsequent executions run quickly. If a plain Statement object is used, the SQL must be compiled and run every time it is executed. Use Extended Dynamic support; it caches the SQL statements in SQL packages on the AS/400 system. Also turn on package cache; it caches SQL statements in memory.

Do not use a PreparedStatement object if an SQL statement is run only one time. Compiling and running a statement at the same time has less overhead than compiling the statement and running it in two separate operations.

Consider using JDBC stored procedures. In a client/server environment, stored procedures can help reduce communication I/Os and thus help improve response time.

Use a just-in-time (JIT) compiler for your Java execution environment if possible. The latest JIT technology allows Java programs to perform almost as well as native code written in C or C++. Users of the AS/400 Toolbox for Java can expect the JDBC driver to perform almost as well as a C++ application using ODBC for OLTP types of applications. JDBC applications that download larger amounts of data will perform slower than a comparable C++ application in ODBC because of the object orientated design of the JDBC driver.

There are many properties that can be specified on the JDBC URL or in the JDBC properties object. Several of these properties can significantly affect the performance of a JDBC client/server application and should be utilized where possible. The properties control record blocking, package caching, and extended dynamic support. See the JDBC driver documentation for details on setting these properties. Most of the properties have close parallels in the ODBC driver. Tuning advice from ODBC can be used for JDBC when setting these values.

Record Level Access

AS/400 physical files can be accessed a record at a time using the public interface of these classes. Files and members can be created, read, deleted, and updated. The record format can be defined by the programmer at application development time, or can be retrieved at runtime by the AS/400 Toolbox for Java support. These classes use the DDM server to access the AS/400 system. To use the host DDM server through a TCP/IP interface, some special PTFs are required. Check the AS/400 Toolbox for Java documentation for the latest PTF numbers and set up instructions.

Record Level Access can offer better performance than JDBC for applications that need to process AS/400 database data one record at a time. Record access does not go through the SQL query processing that JDBC must go through to process data, therefore it can retrieve a single record quicker than JDBC. However, if complex computations or large sets of records are processed, JDBC may be a better solution. Use JDBC when an SQL statement can be built that does most of the work on the server, or when you want to limit which fields in a record get transfered to the client. The current JDBC specification does not have a mechanisim to insert multiple records at a time (e.g., ODBC′s Blocked Insert). Therefore, use the support in Record Level Access to write multiple records at once.

There are several issues that should be considered when using the Record Level Access support in the Toolbox. First, when accessing a file, if the program is going to use the file multiple times, the file should be left open inbetween operations to avoid the extra processing due to open and close. Second, the block size that is specified on the open method should be selected acording to the type of access in the file. Block size is the number of records to download to the client when reading. If multiple records that are relatively close together are going to be retrieved then a block size that can transmit all of the records at once is prefered. However, do not select a block size that will cause a large delay when downloading (e.g., a 1MB download). If the type of access for the file is random, and the records retrieved are not close together in the file, a block size of 1 is prefered. Third, when downloading an entire file that is relatively small, use the readAll method. This method is considerably faster when reading small files, large files (e.g. >1MB) may encounter ″Out of memory″ errors, beacuse the entire file is placed and translated in the client′s memory.

10.3 Client Access/400

Client Access/400 for Windows 3.1 (XK1)

In V4R1 Client Access has a new Windows 3.1 client (named Client Access Enhanced for Windows 3.1) for customers continuing to use Microsoft Windows 3.1 or Windows for Workgroups 3.11 operating systems on their PC desktops. This client delivers enhancements to customers using Windows 3.1 today and offers key technologies from its Windows 95/NT client such as native TCP/IP support. This Enhanced Windows 3.1 client provides a clear transition strategy for customers who decide later to migrate to the Microsoft Windows 95 or NT operating system on their desktop.

In V4R1 Client Access/400 added TCP/IP support to the Windows 3.1 client. This addition opens the Client Access/400 for Windows 3.1 product to run over the internet. TCP/IP is the recommended connectivity to the AS/400 for Windows 95/NT, and provides the best upgrade path from Windows 3.1 to Windows 95/NT.

AnyNet support in the Windows 3.1 client has improved in performance; however, TCP/IP will provide faster response times and require less AS/400 CPU due to the extra overhead needed to support AnyNet. Applications that require AnyNet should experience performance that is slightly less than SNA or TCP/IP.

The Data Transfer tool has been improved to reduce the time it takes to download and upload files from the AS/400. With this tool, users can leverage the power of the AS/400 database and the graphical analysis tools on Windows platforms.

The enhancements to Client Access/400 for Windows 3.1 extend the life of PC′s connected to AS/400s by improving performance and adding new support for TCP/IP. With the added support for TCP/IP, the Windows 3.1 client provides clear upgrade paths to Windows 95/NT.

Client Access/400 for Windows 95/NT (XW1)

In V3R7AF Client Access/400 added support for Windows NT. The product Client Access for Windows 95 has been replaced with Client Access for Windows 95/NT.

Client Access for Windows 95/NT communications support provides excellent performance with native TCP/IP connectivity. Additionally, its native APPC support is equal to or better than most other 32-bit APPC Router implementations available for Windows 95/NT. For optimum client/server application response time performance, we recommend that customers use TCP/IP connectivity.

There are several migration scenarios involved with the AS/400 Client Access for Windows 95/NT client, and each one has different performance characteristics:

1. Customers using APPC connectivity on the AS/400 Client Access for Windows 3.1 client or DOS Extended client may experience a client response time degradation when migrating to AS/400 Client Access for Windows 95/NT *APPC* connectivity. AS/400 server capacity is unchanged.
2. Customers using APPC connectivity on the AS/400 Client Access for Windows 3.1 client or DOS Extended client will in most cases experience equivalent client response times when migrating to native *TCP/IP* connectivity on the Client Access for Windows 95/NT client. AS/400 server capacity may be somewhat reduced.
3. Customers using AnyNet (APPC over TCP/IP) connectivity on the AS/400 Client Access Windows 3.1 client will, in most cases, experience improved client response times when migrating to native TCP/IP connectivity on the Client Access for Windows 95/NT client. Native TCP/IP connectivity is available to all Client Access applications or to applications written to Client Access for Windows 95/NT 32-bit Application Programming Interfaces (API′s). Applications written to Windows 3.1 Client Access API′s, will run over SNA or, in addition, over TCP/IP by using AnyNet (APPC over TCP/IP). AS/400 server capacity will be improved in most cases.
4. Customers migrating from Client Access for Windows 95 to the new product, Client Access for Windows 95/NT, should experience equivalent client response times. Users changing client operating systems from Windows 95 to Windows NT and using the new Client Access/400 may notice a slight improvement in response time for some operations.
5. Slower clients (′486 66MHZ or less) running SNA may experience a performance degradation when migrating to the Client Access for Windows 95/NT from Client Access for Windows 95. The level of degradation depends on the environment. Typical operations using SNA with slower clients should be no more than 10-50% slower than Client Access for Windows 95/NT. Faster clients may experience very little, if any, degradation.

Due to the robust nature of the SNA router connectivity in the Client Access Windows 95/NT client, startup response times may be significant, especially on PC′s with lower amounts of memory. The Windows 95 client performs best on a 486 66 MHZ client or greater system with at least 16 MB of memory. The

Windows NT client performs best on a Pentium 133 MHZ or greater system with at least 32 MB of memory. Refer to Client Access information for the minimum system sizes supported.

After the first connection has been established to an AS/400, subsequent connections to the system or any other system are much faster. Because of this, it is best to leave open connections up, if possible, and not reconnect each time you use an application running over this connectivity.

ODBC users should be aware that a parameter setting has changed with the new Client Access for Windows 95/NT. A PREFETCH setting was added with the default set to OFF. Previously, PREFETCH was built-in and always set to ON. This setting was added because PREFETCH is not supported for applications that use SQLExtendedFetch commands. Some users may see a performance degradation if PREFETCH is not enabled. Users that use SQLExtendedFetch should leave this setting OFF.

OLE/DB and ADO Data Access (Project Lightning)

The OLE/DB driver that has been added to the base Client Access/400 for Windows 95/NT allows database access to the AS/400. Developers can write applications that use the OLE/DB driver to access the AS/400 database through DMM Record Level Access, SQL, Stored Procedures, etc. These interfaces can be easily programmed through the ADO layer that most current development environments support (e.g., Visual Basic, Delphi, etc).

The current ADO specification does not support record blocking; therefore, downloading a large table through record level access may take longer than other methods (e.g., ODBC which has record blocking) and consume more network resources, since each record is transmitted as one communication. Look for future versions of the ADO specification to contain record blocking.

The SQL support in the OLE/DB driver uses the same server program as the ODBC driver. This means that developers can use some of the same techniques and ideas from the Client Access ODBC driver for the OLE/DB driver. One important performance improvement that developers can use, is implement prepared statments if an SQL statement is to be executed more than once. Also when executing a prepared statment set the third parameter to be -1, otherwise ADO assumes this is a new statement and discards the previously prepared statement.

Open Data Base Connectivity - ODBC(**)

In V3R1, the ODBC APIs were significantly enhanced in terms of function and performance compared to the original version in V2R3 (see *AS/400 Performance Capabilities Reference (V3R2), ZC41-8166*, Chapter 7, for more details). ODBC support for the Windows (**) 3.1/95/NT clients in Client Access/400 provides superior performance over the original Remote SQL support in Client Access/400 and its predecessor PC Support/400.

ODBC is a set of API′s (Application Programming Interfaces) which provide clients with an open interface to any ODBC supported database. AS/400 supports ODBC with the Windows 3.1, Windows 95, Windows NT and OS/2 client support in Client Access/400. Customers can purchase ODBC drivers to connect to their system(s) and either write applications which utilize the ODBC APIs or purchase an existing application which utilizes the ODBC APIs.

Client/Server 4GL and Middleware

Many users build client/server applications using client toolkits such as C/S fourth generation languages (4GLs) or CASE (Computer-Aided Software Engineering) tools. Most of the new 4GL tools use ″middleware″ or interface code to connect to a server. In Windows or OS/2 this middleware usually consists of one or more DLLs used to connect to a given server. The middleware converts the client′s request into commands and data which the server can understand and converts the server′s response into commands and data which the client can understand. Often the middleware is written by the toolkit provider to interface to a given server or to a standard server API set.

Examples of 4GL toolkits are GUPTA′s SQLWindows**, PowerSoft′s Powerbuilder**, and Microsoft′s VisualBasic**. Examples of middleware standards are Microsoft′s ODBC standard and IBM′s DRDA standard.

Because the user is often isolated from the APIs and the middleware manages the database access method, it is important to build applications using tools that optimize for performance. In many cases, tools that are built for ″openness″ for many servers tend to be the worst performers because they are built to the least common denominators. The AS/400 supports many features that enhance performance. Ensure that your toolkit has support for functions like stored procedures and blocked insert. If not, ensure that there is a mechanism to write directly to the CA/400 API set for the best performance. For more information on client/server application development tools, see *AS/400 Client/Server Performance Using Application Development Tools*, (SG24-4731).

Client Toolkit ODBC Performance

Many performance problems with client development toolkits are due to the client tool creating inefficient database access requests. For example, a simple database transaction that should result in minimal interaction with the server can generate hundreds of unnecessary ODBC requests and responses. By choosing high performance toolkits and with planning and tuning, these problems can be avoided. Refer to the *Client Access/400 Windows 3.1 Client for OS/400 ODBC User′s Guide* (SC41-3533) for more information on specific client tools.

Tools are available to diagnose and debug problems with client toolkits and applications. Use tools such as ODBCSpy or ODBC Trace (available through the ODBC Driver Manager) to verify the efficiency of the SQL and ODBC calls that are generated. Also, the toolkits themselves often have tools to trace their server access methods.

Client/Server Online Transaction Processing (OLTP)

OLTP applications are typically designed for business computing. An OLTP transaction usually consists of several database operations and related computations. Performance requirements for this type of transaction are usually stringent. In the client/server environment an OLTP transaction typically consists of several requests/responses between the client and the server, resulting in a small to moderate amount of data transferred to the client. Because these transactions tend to be repetitive, they are good candidates for application serving (such as remote procedures or distributed processing). It is especially important to avoid unnecessary overhead in processing these repetitive transactions (such as PREPARE operations). Use of parameter markers, stored procedures, triggers, and Extended Dynamic (package) support is recommended to improve the performance of this class of queries.

Server Challenge Benchmark (SCB)

SCB Overview: The Server Challenge Benchmark (SCB) is a set of three individual workloads developed by IBM to compare the AS/400 against the competition:

1. Transaction-based Component Workload
2. Decision Support Component Workload
3. File Server Component Workload

Each of these component workloads were built to use client/server characteristics. For a detailed description of SCB refer to the IBM white paper titled ″AS/400 Client Server Performance Benchmark Guide″ available on HONE.

Figure 29 shows the performance curves for AS/400 systems running a subset of the Server Challenge Benchmark (SCB). The Online Transaction Processing (OLTP) workload is being used, which is a client/server Order Entry application.

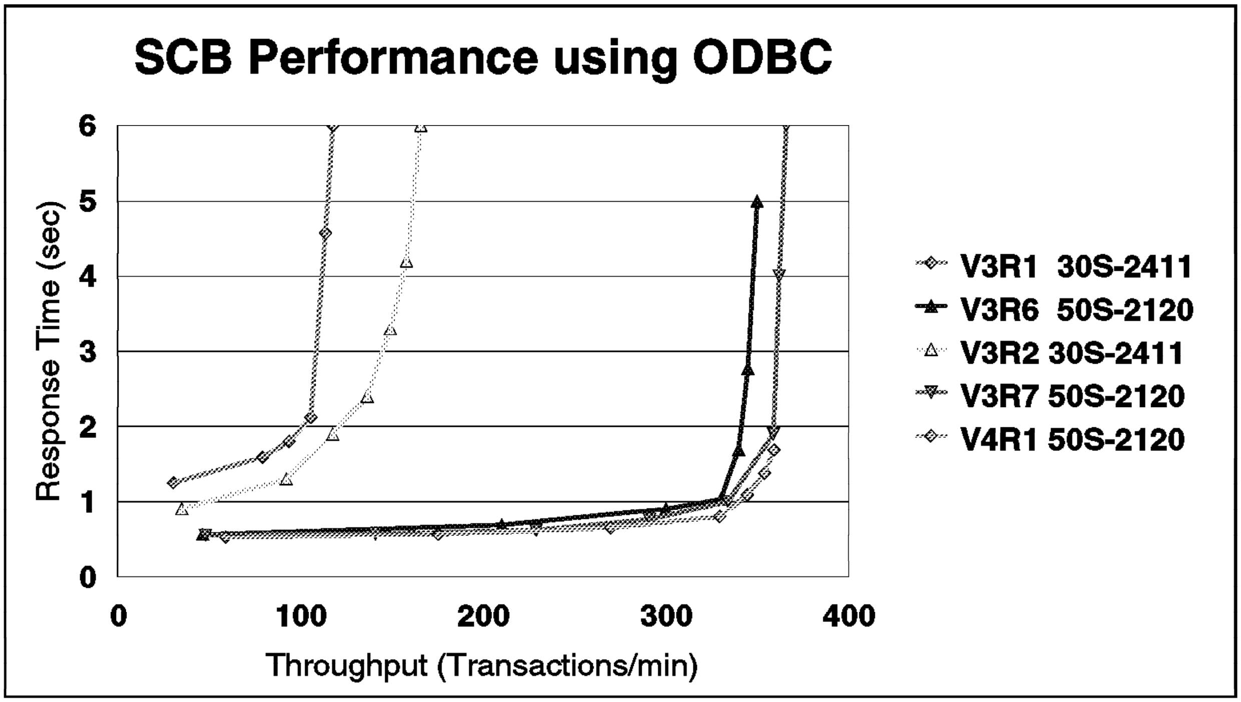


Figure 29. Server Challenge Benchmark OLTP Performance

This application is typical of a commercial client/server application where many small transactions are being continuously processed. It is not representative of a decision support workload nor that of a file server workload, which are the other two components of the SCB. Additional information on file server performance can be found in Chapter 9, “AS/400 File Serving Performance” on page 137.

Figure 29 shows the system performance growth capability offered by the new AS/400 models using PowerPC technology. In this case, the 50S-2120, which is the logical upgrade path for the 30S-2411, provides significant system performance growth for OLTP transaction processing. Actual growth achieved in specific environments will depend on the workload and application characteristics.

Conclusions/Explanations:

1. V3R2 provides significantly better response time than V3R1. This is primarily due to performance improvements in Client Access for Windows 3.1 V3R1M1. The V3R1 measurements were done using CA/400 for Windows 3.1 V3R1M0. Capacity improvements were gained through server and database optimizer improvements.
2. V4R1 provides slightly better throughput/capacity than V3R7. this is due to improvements in SLIC operating system components and database optimizer improvements.

SCB with Windows 3.1/95/NT Client

Measurement Configuration:

AS/400 50S - 2120 - dedicated
V4R1
256 MB Memory
2-6606, 2-6605 DASD (5.99 GB)
Ballista IOP
Clients:
o Win 3.1 client - Pentium-133 MHZ - 32MB memory - Windows 3.1
CA/400 for Windows 3.1 XK1
o Win 95/486 client - ValuePoint 486-66 MHZ - 32MB memory - Windows 95
CA/400 for Windows 95/NT V3R1M2
o Win 95 client - Pentium-133 MHZ - 32MB memory - Windows 95
CA/400 for Windows 95/NT V3R1M2
o Win NT client - Pentium-133 MHZ - 32MB memory - Windows NT 4.0
CA/400 for Windows 95/NT V3R1M2
16 Mbps TR-LAN
SCB Transaction-based Workload - implemented with Visual C++
One client - Average transaction response time - 100 transactions
(All response times are expressed in seconds)

Measurement Results:

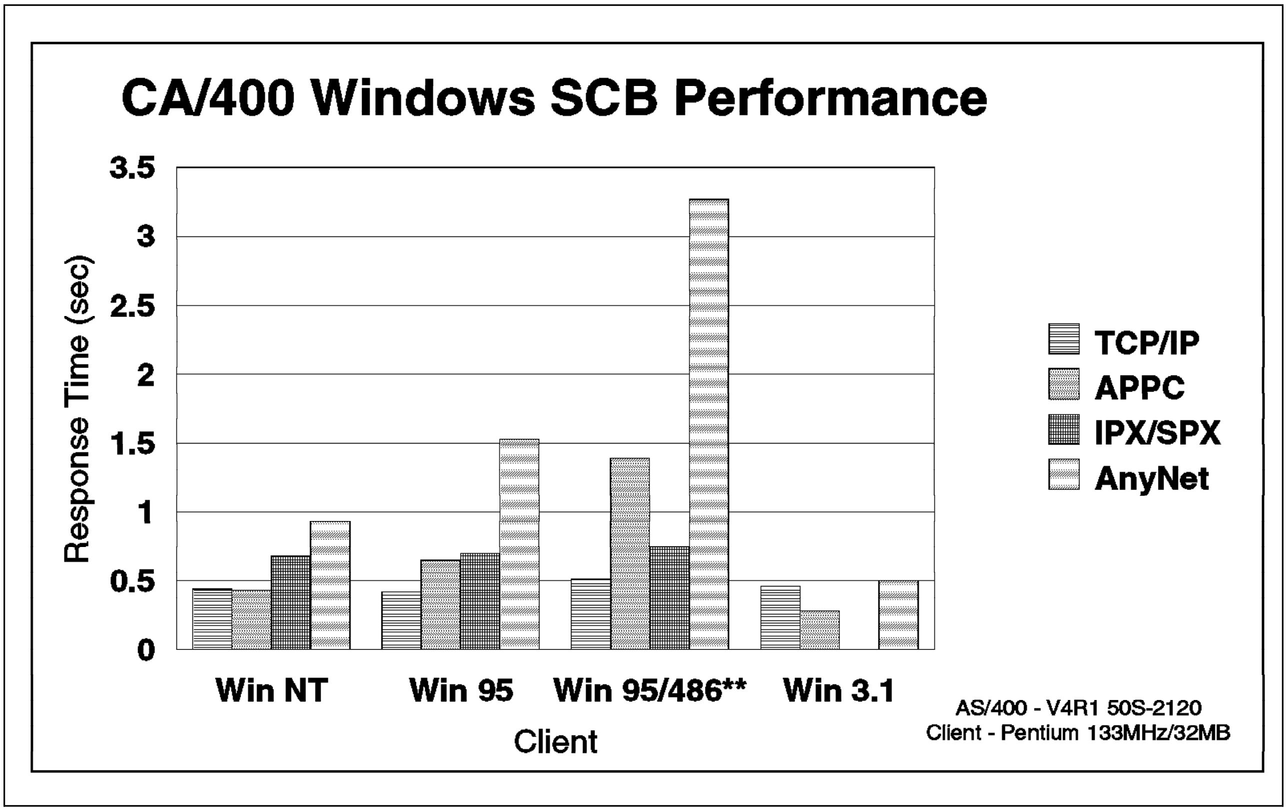


Figure 30. CA/400 Windows SCB Performance

Conclusions/Explanations:

1. Windows 95/NT clients should use TCP/IP for the fastest response time. TCP/IP response time is fastest because the client response time is best with TCP/IP. Note that AS/400 server time is slightly higher for TCP/IP than APPC.

2. Windows 3.1 clients will achieve the best response times with APPC. However, TCP/IP can provide equal or better response times in different situations. TCP/IP is also the strategic connectivity for the AS/400; it provides the best upgrade path to Windows 95/NT, and it is a step to enabling applications for the internet. TCP/IP support was added to the Windows 3.1 client in V4R1 as the XK1 product.

3. Windows NT clients have better APPC response times than Windows 95 because of the architectural differences between the two client operating systems.

4. CA/400 clients using AnyNet should experience an improvement in response time and capacity when migrating to native TCP/IP. Client and server contributions to response time are faster for TCP/IP than AnyNet.

5. CA/400 for Windows 3.1 clients using APPC may experience a degradation in response time when migrating to CA/400 for Windows 95/NT. This is due to client architecture differences between Windows 3.1 and Windows 95/NT. AS/400 server performance is unchanged between Windows 3.1 and Windows 95/NT clients; therefore, AS/400 server capacity is unchanged.

6. Clients directly connected to the AS/400 server will generally achieve fastest response times. Gateways tend to add response time to client/server transactions. Performance-sensitive workloads usually perform best when

directly-attached to the AS/400 server since the number of processing layers is minimized.

AS/400 vs Client Time for SCB
Measurement Configuration:

AS/400 50S - 2120 - dedicated
V4R1
256 MB Memory
2-6606, 2-6605 DASD (5.99 GB)
Ballista IOP
Client:
o Win 95 client - Pentium-133 MHZ - 32MB memory - Windows 95
CA/400 for Windows 95/NT V3R1M2
16 Mbps TR-LAN
SCB Transaction-based Workload - implemented with Visual C++
One client - Average transaction response time - 100 transactions
(All response times are expressed in seconds)

Measurement Results:

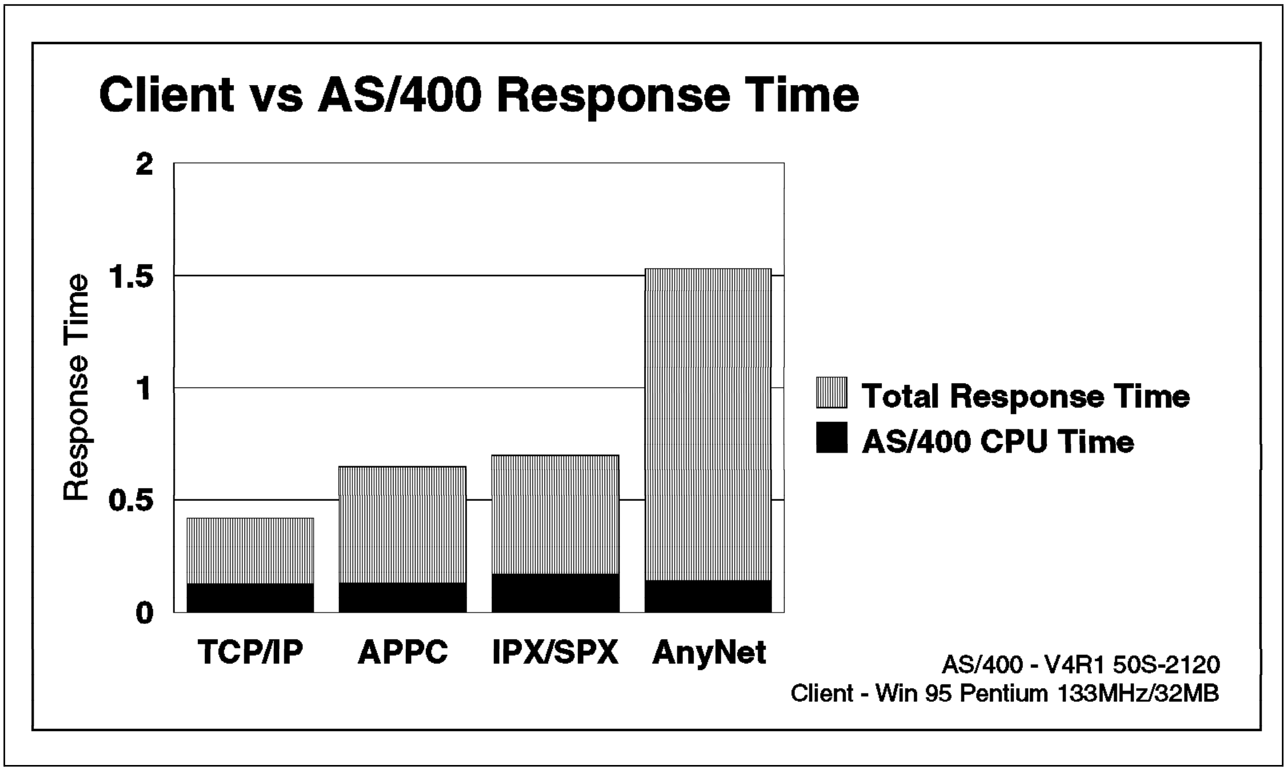


Figure 31. Client vs AS/400 Response Time

Conclusions/Explanations:

1. Response time performance of most client/server workloads can be improved by using faster clients since, for most workloads, a significan portion of the overall workload is spent on the client. As shown in the data above, no more than half of the SCB workload response time is due to the AS/400 CPU.

2. AS/400 server capacity is best with APPC. This is due to the AS/400 server communications pathlength difference between TCP/IP and APPC. However this difference has been reduced in V4R1, the difference is now less.

SCB Scaling using ODBC

Measurement Configuration:

```
Servers:
 o AS/400 50S - 2120 - dedicated
   V4R1
   256 MB Memory
   DASD 5.99GB, 4 arms
   Ballista Token Ring IOP - 6149
 o AS/400 640 - 2239 - dedicated
   V4R1
   8 GB Memory
   DASD 549 GB, 131 arms
   10 Ballista Token Ring IOP - 6149
Clients:
 o 16 Win NT clients - Pentium-133 MHZ - 128MB memory - Windows NT
                   CA/400 for Windows 95/NT V3R1M2
16 Mbps TR-LAN
SCB Transaction-based Workload - implemented with load simulation tool
 o Average transaction response time
 o Average transactions per second
 o 20 minute runs
(All response times are expressed in seconds)
```

Measurement Results:

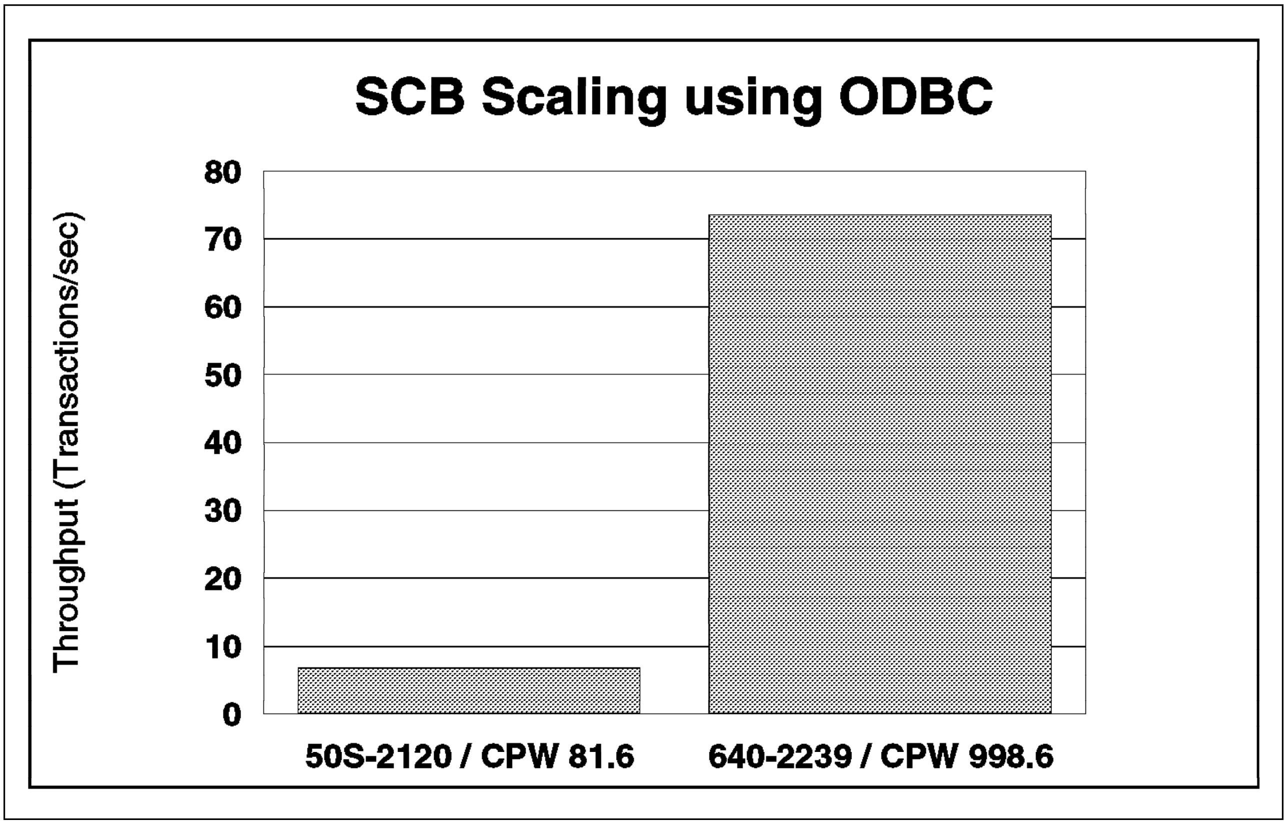


Figure 32. SCB Scaling with ODBC

Conclusions/Explanations:

1. Applications that use ODBC properly can scale well when run to AS/400s. Application developers should be careful to create programs that minimize locking on shared resources or do not consume large portions of the system. These situations will inhibit programs from scaling on any system.
2. The AS/400 product line provides a clear upgrade path for solutions that grow over time. With the introduction of the high-end n-way systems in V4R1, AS/400s have proven that system capacity will grow proportional to CPWs.
3. Application performance may vary from the results shown here. Application scalability depends upon resources beyond just CPU. All aspects of the system should be taken into account. Program design can be the largest factor in application scalability.

Client/Server Query and Decision Support

Query and Decision Support (also known as ad-hoc queries) are database operations typically done throughout the day in many businesses. These operations are longer-running server-intensive database operations which are usually read-only. Unlike OLTP, they are seldom done frequently and although throughput may not be a critical factor in performance, response time surely is. These queries usually result in minimal interactions between the server and the client. Although they may return a large number of rows of information, typically ad-hoc queries return few rows. Because these queries are server-intensive, they are good candidates for database serving. The response time of these remote (c/s) queries are not significantly slower (typically 5% - 30%) than local (host terminal) queries. However, since the server performance load can be large, the user may benefit from moving the execution of these queries off-shift.

In cases where the query executed is CPU-intensive and very few records are returned to the client, the response time of the query is typically very close to running the same query interactively on the server.

ODBC Query Using Re-use Pre-started Jobs

V3R7 OS/400 provides a new option to re-use pre-started jobs for the QSERVER subsystem. This allows pre-started jobs to ″recycle″ jobs that have previously ended. As a result, CPU time to start-up jobs is reduced with the re-use value set to greater than 1.

This value can be changed with the AS/400 command CHGPJE. Use the QSERVER subsystem with the job QZDAINIT and subsystem QIWS for APPC jobs or QZDASOINIT for TCP/IP or IPX jobs. Press F10 for additional parameters then page down to the parameter ″MAXIMUM NUMBER OF USES″. Enter the new value for the maximum job re-uses. The V3R7 default value is 200.

Query Download/Upload (Database file transfer)

Download/Upload queries represent a set of queries that either fetch a significant number of rows from DB2/400 tables (or files) or insert a significant number of rows into DB2/400 tables or files. Because of the number of rows processed, there is a significant amount of processing that occurs on the client. Many times, significant performance gains may be realized by running these types of queries on a faster client processor.

Query Download Comparisons

This section compares the performance of downloading a significant number of records from an DB2/400 database file into a client application using various CA/400 APIs. To obtain the download comparisons, a client application was developed that fetches about 1.4 MBs of data using different APIs. All fetched character data was converted from EBCDIC to ASCII automatically by CA/400 functions. No further processing was performed on the data retrieved.

Measurement Configuration:

The following configuration was used to perform the query download measurements:

```
AS/400 Server 50S-2120 - dedicated
256 MB Memory
2-6606, 2-6605 DASD (5.99 GB)
ValuePoint clients - ′486-66 MHZ - 32MB memory - CA/400 for Windows 3.1 V3R1M1
2048 byte frame size
16 Mbps TR-LAN
```

The download operations were done using standard PC-based query tools. The tool was changed to display the first resulting rows or to simply indicate completion of the test. For query download tests, the ODBC.INI file on the client was changed to vary the Record Blocking size setting. The following queries were performed to download records from the DB2/400 database files:

Table 47. CA/400 Windows 3.1 Query Download Tests

Query	SQL Statement	Row Size	Columns per Row	Rows Fetched	Bytes Fetched
LBR	SELECT * FROM DBITRK/LBRSTATS	118	10	12,000	1,416,000
TRK	SELECT * FROM DBITRK/TRKOPRNS WHERE TRKITEMN <′ITE02210MNUMBR′	314	42	4,418	1,387,252

Measurement Results:

The following table shows the ODBC.INI Record Blocking size setting, the overall download rates in megabytes per hour, the overall response times in seconds, and the AS/400 CPU seconds consumed for the queries listed above. The queries were implemented using Client Access/400 ODBC.

Table 48. CA/400 Windows 3.1 Query Download Performance

Query	Record Block Size	Transfer Rate (MB/hr)	Response Time (seconds)	AS/400 CPU consumed (sec)
LBR - display rows	512K	296	17.2	0.63
LBR - no display	512K	566	9.0	0.63
LBR - no display	32K	520	9.8	0.69
TRK - display rows	512K	271	18.4	0.78
TRK - no display	512K	450	11.1	0.78
TRK - no display	32K	427	11.7	0.86
Note: All tests received all rows; ″display″ tests only displayed the first set of rows				

Query tools are available to provide the client with an easy, graphical way to access server databases. For performance reasons, many of these tools allow the user to limit the number of rows received. We ran one of the above queries and limited the rows using a query tool. This table shows the row limit, the ODBC.INI Record Blocking setting, and the overall response times in seconds.

Table 49. CA/400 Windows 3.1 Query Download Performance - Limited Rows

Query	Row Limit	Record Block Size	Response Time (seconds)
LBR	20	32K	1.4
LBR	20	512K	3.3
TRK	20	32K	2.2
TRK	20	512K	4.1
Note: These tests only downloaded the first 20 rows of data			

For comparison, we measured an IFS file transfer (download) operation using one of the same database tables above. We converted the DBITRK/LBRSTATS table to a stream file and loaded it on the server in the IFS root directory. We then used Windows File Manager to copy the file from the server to the PC hard disk. We compared the time for downloading a new file with replacing an existing file. The following table shows the overall download rates in megabytes per hour, the overall response times in seconds, and the AS/400 CPU seconds consumed for the queries listed above.

Table 50. CA/400 Windows 3.1 File Download Performance

File Downloaded	File Transfer Rate (MB/hr)	File Transfer Time (seconds)	AS/400 CPU consumed (sec)
DBITRK/LBRSTATS - new file	593	8.6	0.59
DBITRK/LBRSTATS - replace file	614	8.3	0.51

Note: These tests downloaded the file to the PC hard disk

Conclusions/Recommendations:

1. Only a small percentage of the total transfer time was due to the AS/400 CPU. Most of the time spent for large record download operations is in the client and communications time. Use fast clients for the best performance. Use fast communications adapters for higher throughput.
2. ODBC query download rates can be comparable to IFS file transfer rates.
3. For fastest retrieval times for an entire large database table, do not immediately format and display all the data retrieved. Instead, use client tools to manipulate and display the data after it has been entirely downloaded to the client.
4. When retrieving the entire database table, the recommended ODBC Record Blocking setting is 512 KB. Decreasing this size may cause slower performance. Memory-constrained clients may require a smaller block setting.
5. When using client tools to browse through the data, limit the query to display only the first screen of data. Fetch the next set of data when needed. Set the Record Blocking to 32K or less for fast retrieval of only a small number of rows from a large table.
6. As the number of columns to be retrieved increases, the retrieval rate decreases and response time increases.
7. The token-ring frame size used was 2K. Larger frame size settings may improve performance.

Query Upload Scenario

This section compares the performance of uploading a significant number of records into a DB2/400 database file from a Windows 3.1 application using various CA/400 ODBC APIs.

Measurement Configuration:

The following configuration was used to perform the query upload measurements:

```
AS/400 Server 50S-2120 - dedicated
256 MB Memory
2-6606, 2-6605 DASD (5.99 GB)
ValuePoint clients - ’486-66 MHZ - 32MB memory - CA/400 for Windows 3.1 V3R1M1
2048 byte frame size
16 Mbps TR-LAN
```

The client application is written in C and utilizes CA/400 Windows 3.1 ODBC APIs to do single inserts and blocked inserts to a table within the DB2/400 database.

The following table gives a brief description of the SQL statements issued and the row descriptions. Note that the questions marks ("?") within the statements are parameter markers or variables that the client application supplies to the ODBC APIs.

Table 51. CA/400 Windows 3.1 Query Upload Tests

SQL Statement	Row Size	Columns per Row	Rows Inserted	Bytes Inserted
INSERT INTO JMBCOLL.PERF VALUES (?,?,?,?,?,?,?,?,?,?)	100	10	1,000	100,000
INSERT INTO JMBCOLL.PERF 350 ROWS VALUES (?,?,?,?,?,?,?,?,?,?)	100	10	30,450	3,045,000

Measurement Results:

The following table shows the overall upload rates in megabytes per hour for the above SQL statements. The single insert case sends 1000 ODBC SQLExecute commands to the AS/400 server to perform 1000 inserts while the blocked insert scenario sends 87 ODBC SQLExecute commands to the server to perform 30,450 inserts.

Table 52. CA/400 Windows 3.1 Query Upload Performance

SQL Insert Row Count	Rows Inserted	Insert Rate (MB/hr)
1	1000	16
Block of 350	30,450	215

Conclusions/Recommendations:

1. Use blocked insert when possible

   Client applications that perform inserts, updates or deletes will generally perform these SQL commands one at a time to the CA/400 data access server. However, for inserts, there is an opportunity to use the blocked INSERT SQL statement which can be used to send a set of rows to the server in a single communications flow. Measurements have demonstrated that this form of insert can be over 20X faster than doing inserts one at a time.

2. Use faster clients

   A large portion of upload and download operations is due to the client. Increasing the speed of the client can improve throughput.

3. Use faster communications adapters

   Using slower communications adapters can result in costly delays. Upgrading the communications adapters can improve throughput.

Client Access/400 for Windows 95/NT 5250 Emulator Performance

This section shows the performance of 5250 emulation using Client Access for Windows 95/NT compared to a 5250 terminal.

Client Access provides the capability to emulate 5250 terminal sessions with the flexibility to configure the keyboard and display to the user′s preferences.

5250 Emulator Performance Results

Measurement Configuration:

AS/400 200-2030 - dedicated
V3R2
16 MB Memory - 4-6609 DASD (8 GB)
Clients:
  o ValuePoint client - ′486-66 Mhz - 32MB memory - Windows 95
  o Pentium client - Pentium-133 Mhz - 32MB memory - Windows 95
Compared to 5250 display: 5291 model 2
CA/400 for Windows 95/NT V3R1M2
16 Mbps TR-LAN
Typical OS/400 5250 workstation screens - 24x80 resolution

Measurement Results:

Table 53. CA/400 Windows 95/NT 5250 Emulator Performance

5250 Emulator Performance Comparison
CA/400 for Windows 95/NT V3R1M2
AS/400 - V3R2 200-2030
Client - ′486 @ 66 MHZ versus Pentium @ 133 MHZ versus 5250 display

Workstation screen	Win-95 ′486 Resp Time (seconds)	Win-95 Pentium Resp Time (seconds)	NT Pentium Resp Time (seconds)	5250 Display Resp Time (seconds)
WRKACTJOB	0.86	0.79	0.72	0.63
WRKLIB *ALL	0.80	0.70	0.69	0.59
WRKOBJ *FILE	0.63	0.55	0.46	0.38

Note: Average response time (seconds)

Conclusions/Explanations:

1. Client Access/400 for Windows 95/NT provides good 5250 emulator performance. NT provides slightly faster performance than Windows-95.
2. Faster clients provide faster response time. The Pentium times provide response times closer to the 5250 display response times.

10.4 Tips for Improving C/S Performance

Following are some tips to use when writing a client/server application which will provide the best performance:

1. Choose the right client processor.

   In many queries the majority of response time is due to client processing, especially when utilizing client/server tools (eg. 4GL/CASE tools). If response time is critical, choose a fast client processor. Figure 28 on page 166 shows the potential increase in response times for queries relative to performing the queries on a ′486 @ 66MHZ client:

2. Ensure that the client is optimized for performance

   Client/server applications usually require a large amount of processing power for both the client and the server. The client processor speed must be appropriate for the task. Also, memory requirements for the client may be large (toolkits, communications -- routers, buffers, etc., and operating system requirements). The response time variation between a fast well-tuned client and a under-powered client can be astounding.

3. Use the fastest communications media possible and tune the communications configuration settings

   Most client/server applications tend to send a large number of requests and responses between the client application and the server. To minimize the delay due to this communications traffic, use fast media such as local area networks (LANs) to attach clients to the server. If response times are critical, do not use wide-area networks (WANs) since the communications speeds are typically measured in thousands of bits per second instead of millions of bits per second. Also, be wary of bridges, routers, and gateways since they may introduce delays when communicating across networks. Instead, keep the response time-critical clients on the same network as the server.

   Use a fast PC communications adapter -- especially for file transfer operations. The communications adapter can be a major factor in constrained throughput. For download operations, a slow communications adapter can reduce throughput by over 10X compared to a fast adapter. The adapter can not keep up with the server and this results in overruns. When an overrun occurs, the server must detect the error and resend the data. This can result in large delays. If a faster adapter can not be used, communications can be tuned to reduce overruns. Following are some examples:

   - TCP/IP for Windows 95/NT

     When using the native Windows 95 TCP/IP communications stack, a registry entry can be changed to improve problems when using slow adapters or slow PCs. This setting can reduce performance on fast adapters and should only be changed when client adapter problems exist.

     ″HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VxD\MSTCP″

     Use REGEDIT to add a new string value ″DefaultRcvWindow″. Set the value to 4096 or decrease until retries are reduced.

     When using Windows NT, the value to change is: ″HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Tcpip\Parameters″

Use REGEDT32 to add the new REG_DWORD value ″TcpWindowSize″. Set the value to 4096 or decrease until retries are reduced.

- APPC

  Set the LANWDWSTP setting from the default of 0 to 2 or greater. For slow adapters, this will reduce the time to correct data re-transmission problems.

- Netsoft APPC Router

  Consider increasing the parameter MAXDATA size from the default value of 521 to the maximum size. This value is specific to each router and can be different for each router configured. The MAXDATA size must be equal to or less than the frame size opened for the network adaptor. Increasing this value can improve performance, in particular the performance of large data transfers.

  To change this parameter, open the ′Netsoft Administrator′ folder, select ′Set Properties′ of specific AS/400 configuration. Next select ′Properties′ of the link being used (for instance 802.2). Finally, select the ′Advanced′ tab.

- IPX/SPX

  If you are using IPX/SPX for large file transfers the default data size sent to IPX/SPX may be increased. Create the string value:

  ″HKEY_CURRENT_USER\Software\IBM\Client Access\CurrentVersion\
  Internal_Components\Communication\Environments\(Your Env)\(Your
  System)\ IPX Max Send

  The default is 1400, the maximum is 65536. Setting this value above the default may cause errors in some configurations. If problems appear after changing this value, delete the registry entry.

The frame size and buffer size for your network card should be increased to optimize the network traffic for your situation. Large file transfers perform better with larger frame sizes if your network adapter and network devices support the larger sizes. Increased buffers allow the client to offload more work to the network card. These settings can usually be controlled through the Control Panel/Network/Adapter Properties, check with your network adapter manufacturer for details. Increasing these setting will increase the system resources used by your network adapter.

Consider the following tuning tips for AS/400 communication.

- Consider increasing the Maximum Transmission Unit (MTU) from the default value of 576. The AS/400 defaults to 576 when a route is added to the configuration (via CFGTCP option 3). This value ensures packets will not be dropped over this route as all TCP/IP implementations have to support at least a 576 byte Transmission Unit.

  In many cases, this value is unnecessarily small. For instance, if the route will only be used on the configured Ethernet or Token Ring, and there are no intermediate hops that only support a 576 byte packet. If this is the case, change the Route Maximum Transmission Unit size to *IFC. This will change the MTU on the Route to the Interface MTU size which defaults to the Line Description Frame size. This defaults to approximately 2000 for Token Ring and 1500 for Ethernet.

- Consider increasing the TCP receive buffer size from the default size of 8192 bytes to a larger value, for example 64384 bytes (via CFGTCP option 3).

  This value specifies the amount of data the remote system can send before being read by the local application. If there are many retransmissions occurring due to the overrunning of a network adapter, decreasing this value instead of increasing it, could help performance.

- Consider increasing the TCP send buffer size from the default size of 8192 bytes to a larger value, for example 64384 bytes (via CFGTCP option 3).

  This value provides a limit on the number of outgoing bytes that are buffered by TCP. If there are many retransmissions occurring due to the overrunning of a network adapter, decreasing this value instead of increasing it, could help performance.

- Refer to section Chapter 5, “Communications Performance” on page 79 for AS/400 communication tuning guidelines and specifically section 5.2, “LAN Protocols, Lines, and IOPs” on page 82

ANYNET support allows clients to run APPC based applications over TCP/IP. ANYNET can be considerably slower than TCP/IP and consumes more CPU than TCP/IP. Client Access for Windows 95/NT allows clients to access the AS/400 directly through TCP/IP. TCP/IP provides faster response times than ANYNET.

4. Ensure that all database requests are optimized for performance

   Although the AS/400 database manager does a good job of handling database requests, it is important that performance-sensitive operations be tuned for optimal performance. Examples of database tuning are: ensure that indexes are being used, simplify SQL statements, minimize redundant operations (re-preparing SQL statements, etc.), and reduce the number of communications requests/responses by blocking. Use tools such as communications trace (STRCMNTRC), DB2/400 debug (STRDBG), Explain function (PRTSQLINF), and Performance Monitor (STRPFRMON) to assist with performance tuning.

5. Ensure that the database access method is tuned for performance

   Whether the application uses DRDA, DDM, Remote SQL, DAL, or ODBC, tuning the access method can result in performance improvement. Client Access/400 ODBC support allows each client to customize the ODBC interface to the AS/400. This is done using the ODBC.INI file for Windows 3.1 or the ODBC Adminstrator for Windows 95/NT. For ODBC tuning guidelines, see section “ODBC Performance Settings” on page 188.

6. If a CASE/4GL toolkit is used, tune the application for performance

   Client toolkits can provide large improvements in application development productivity. But, since most are developed to communicate with multiple servers, they may not be optimized for any specific server. For more information on CASE/4GL toolkits, see section “Client/Server 4GL and Middleware” on page 172.

7. Use parameter markers support when performing repetitive transactions

   A parameter marker is a question mark (?) that appears in the SQL statement where a host variable could appear if the statement API string was a static SQL statement. Parameter markers enhance performance by allowing a user to prepare a statement once and then execute it many times using a different set of values for the parameter markers.

8. Reuse prepared statements

   Prepares of SQL statements can take over 0.5 seconds to process. There are two ways to reuse prepared statements:

   - Only prepare statements once (using parameter markers) and use SQLExecute ODBC API

     Reducing redundant prepare statements and using parameter markers instead of literals are two of the best ways to improve database server performance -- especially OLTP operations which are frequently repeated. Response time of a complex, repetitive transaction can be reduced by over 5X by changing the client application to take advantage of these improvements.

   - Use package support

     Package support, available with CA/400 ODBC, provides built-in reuse of prepared statements. See “ODBC Performance Settings” on page 188 for more information on configuring for package support.

9. Use stored procedures and triggers to reduce communication flows

   To reduce network traffic between the client and the server and reduce response time, use stored procedures and/or triggers. Typical database serving applications send or receive from a dozen to a hundred requests/responses. Stored Procedures and triggers can reduce the number of flows significantly. Also, more processing is done at the server so the application can be completed more efficiently.

10. When possible, use SQLExecDirect for one-time execution (one flow, not two)

    SQLExecDirect can replace the pair: SQLPrepare and SQLExecute. **However**, if you are doing multiple executions of the SQLStatement (looping), you should separate the SQLPrepare and SQLExecute such that the SQLPrepare is done only once and the SQLExecute is processed multiple times. This reduces both AS/400 and client processing time because the PREPARE/DESCRIBE steps do not need to be repeated. This is much more efficient than SQLExecDirect.

11. Ensure that each statement has a unique statement handle

    Sharing statement handles for multiple sequential SQL statements causes DB2/400 to do FULL OPEN operations since the database cursor can not be re-used. By ensuring that an SQLAllocStmt is done before any SQLPrepare or SQLExecDirect commands, database processing can be optimized. This is especially important when a set of SQL statements are being executed in a loop. Allowing each SQL statement to have its own handle reduces the DB2/400 overhead.

12. Utilize blocking

    - Use ″FOR FETCH ONLY″ and avoid ″UPDATE WHERE CURRENT OF″
    - Set maximum frame size > 2K for large upload or download

      For the Windows 3.1 client, use the Global Options settings in Configuration to set the maximum frame size. For the Extended DOS client, use the TRMF setting in CONFIG.PCS.

    - Use blocked inserts

Blocked Insert allows a client application to send a set of rows to the server (instead of one at a time). Measurements show that the performance of Blocked Insert can exceed 10X improvement over single row insert (eg. 1000 100-byte rows inserted)

13. Use lowest level of commitment control required

    More server processing is required to process more stringent commitment control settings.

14. Use ODBC instead of Remote SQL

15. Define client column parameter marker variables identical to host column descriptions to allow for direct mapping on the server.

    This reduces the overhead of variable type mapping.

16. Consider tuning some CASE/4GL applications (changing ODBC APIs)

    Customizing ″open″ client applications by using the tips listed above, you may be able to improve overall performance.

17. Choose a server access method which provides high performance database serving

    If your 4GL supports multiple access methods to the AS/400 server, consider the following:

    a. Use ODBC for best SQL access performance

       ODBC can improve performance over other SQL access methods. ODBC is the strategic database serving interface to AS/400.

    b. DRDA

       Distributed Relational Database Architecture (DRDA) provides acceptable performance in most cases. When possible, use static SQL statements for the best performance.

    c. DDM

       Distributed Data Management (DDM) does not have the flexibility of SQL but, in most cases, provides good record-level file access performance.

18. Use client tools to assist in tuning the client application and middleware. Tools such as ODBCSpy and ODBC Trace (available through the ODBC Driver Manager) are useful in understanding what ODBC calls are being made and what activity is taking place as a result. Client application profilers may also be useful in tuning client applications and are often available with application development toolkits.

19. When possible, avoid extra communications layers such as AnyNet for the best performance of OLTP and large record upload/download workloads. Functions that do not require fast response times through the communications layers (e.g., ad-hoc queries and stored procedures) are a better fit for Anynet.

ODBC Performance Settings

You may be able to further improve the performance of your ODBC application by editing the ODBC.INI file on Windows 3.1. The settings in the ODBC.INI file are stored in the registry for Windows 95/NT. The recommended way to access these settings is through the ODBC Administrator in the Control Panel. The settings can be found in the registry under the Key ″HKEY_CURRENT_USER\Software\ODBC\ODBC.INI″. The ODBC.INI file for

Windows 3.1 clients contains information relating to the various ODBC drivers and data sources and is located in the Windows subdirectory for each CA/400 ODBC client. Listed below are some of the parameters that you can set to better tune the performance of the Client Access/400 ODBC Driver. The ODBC.INI performance parameters that we will be discussing are:

- Prefetch
- ExtendedDynamic
- RecordBlocking
- BlockSizeKB
- LazyClose
- LibraryView

Prefetch = (choices 0,1): The Prefetch option is a performance enhancement to allow some or all of the rows of a particular ODBC query to be fetched at PREPARE time. This option is set OFF by default. We recommend that this setting be turned ON. However, if the client application uses EXTENDED FETCH (SQLExtendedFetch) this option should be turned OFF.

ExtendedDynamic = (choices 0,1): Extended dynamic support provides a means to ″cache″ dynamic SQL statements on the AS/400 server. With extended dynamic, information about the SQL statement is saved away in an SQL package object on the AS/400 server the first time the statement is run. On subsequent uses of the statement, CA/400 ODBC recognizes that the statement has been run before and can skip a significant part of the processing by using the information saved in the SQL package. Statements which are cached include SELECT, positioned UPDATE and DELETE, INSERT with subselect, DECLARE PROCEDURE, and all other statements which contain parameter markers.

All extended dynamic support is application based. This means that each application can have its own configuration for extended dynamic support. Extended dynamic support as a whole is controlled through the use of the ExtendedDynamic keyword. If the value for this keyword is 0, no packages are used and no additional information will be added to the ODBC.INI file. If the value is set to 1 (default), when an application is run for the first time, the ODBC driver will add a line to the ODBC.INI file for the datasource in use that looks like this:

```
Package<Appname> = lib/packagename,usage,pkg full option,pkg not used option
```

Once this entry is added to the ODBC.INI file it can be modified to provide the support that the user wants.

Packages may be shared by several clients to reduce the number of packages on the AS/400 server. For the clients to share the same package, the default libraries of the clients must be the same and the clients must be running the same application. Extended dynamic support will be deactivated if two clients try to use the same package but have different default libraries. In order to reactivate extended dynamic support, the package should be deleted from the AS/400 and the clients given different libraries to store the package in. The location of the package is stored in the ODBC.INI file for Windows 3.1 and in the registry for Windows 95/NT.

Usage (choices 0,1,2): The default and preferred performance setting (2) enables the ODBC driver to use the package specified and adds statements to the package as they are run. If the package does not exist when a statement is being added, the package is created on the server.

Considerations for using package support: It is recommended that if an application has a fixed number of SQL statements in it, a single package be used by all users. An administrator should create the package and run the application to add the statements from the application to the package. Once that is done, configure all users of the package to not add any further statements but to just use the package. Note that for a package to be shared by multiple users each user must have the same default library listed in their ODBC library list. This is set by using the ODBC Administrator or by changing the ODBC.INI file.

Multiple users can add to or use a given package at the same time. Keep in mind that as a statement is added to the package, the package is locked. This could cause contention between users and reduce the benefits of using the extended dynamic support.

If the application being used has statements that are generated by the user and are ad hoc in nature, then it is recommended that each user have his own package. Each user can then be configured to add statements to their private package. For each user to have a private package, the ODBC.INI file must be modified so that each user has a different package name. Either the library name or all but the last 3 characters of the package name can be changed.

RecordBlocking = (choices 0,1,2): The RecordBlocking switch allows users to control the conditions under which the driver will retrieve multiple rows (block data) from the AS/400. The default and preferred performance setting (2) will enable blocking for everything except SELECT statements containing an explicit ″FOR UPDATE OF″ clause.

BlockSizeKB = (choices 1 thru 512): The BlockSizeKB parameter allows users to control the number of rows fetched from the AS/400 per communications flow (send/receive pair). This value represents the client buffer size in kilobytes (1kb=1024) and is divided by the size of one row of data to determine the number of rows to fetch from the AS/400 server in one request. The primary use of this parameter is to speed up queries that send a lot of data to the client. The default value 32 will perform very well for most queries. If you have the memory available on the client, setting a higher value may improve some queries.

LazyClose = (choices 0,1): The LazyClose switch allows users to control the way SQLClose commands are handled by the Client Access/400 ODBC Driver. The default and preferred performance setting (1) enables Lazy Close. Enabling LazyClose will delay sending an SQLClose command to the AS/400 until the next ODBC request is sent. If Lazy Close is disabled, a SQLClose command will cause an immediate explicit flow to the AS/400 to perform the close. This option is used to reduce flows to the AS/400, and is purely a performance enhancing option.

LibraryView = (choices 0,1): The LibraryView switch allows users to control the way Client Access/400 ODBC Driver deals with certain catalog requests that ask for all of the tables on the system. The default and preferred performance setting (0) will cause catalog requests to use only the libraries specified in the default library list when going after library information.

Setting the LibraryView value to 1 will cause all libraries on the system to be used for catalog requests and may cause significant degradation in response times due to the potential volume of libraries to process.

AS/400 Memory Requirements

Multiple clients running the CPW workload were used to help determine the optimal amount of AS/400 memory needed per client. It was found that in the range of 2.5 to 2.8 MB per user, client response times ″leveled off″ such that more memory did not significantly improve response times. However, as we continued to add memory to the pool beyond 2.8 MB per user, the paging and faulting for that pool continue to decrease significantly until about 3.2 MB per user was available for each client. See "Client/Server Online Transaction Processing (OLTP)" on page 172 for more information on the CPW workload.

The effects of memory depends on the kind of workload done by the jobs in the shared pool. Your memory requirements may vary depending on many factors such as high communications I/O, accessing very large database tables, frequent DASD I/O accesses, and high level of multi-processing (sharing) in the pool.

Note that the AS/400 Work Management manual (SC41-3306) has a good set of recommendations on memory tuning. Knowledge of these tips can assist in building high-performance system environments.

Chapter 11. Domino for the AS/400

Performance information for Lotus Domino for AS/400 is included in this section. Domino for AS/400 provides a variety of functions, with this section limited to the performance of the mail server function.

There are many factors that can impact overall performance (e.g., end-user response time, throughput) in the AS/400 Domino environment, some of which are listed below:

- AS/400 processor speed
- utilization of key AS/400 resources (CPU, IOP, memory, disk)
- object contention (e.g. mutex waits, lock waits)
- speed of the communications links
- congestion of network resources
- processing speed of the client system

The primary focus of this section will be to discuss the performance characteristics of the AS/400 as a server in a Domino environment, providing capacity planning information and recommendations for best performance. Please refer to Chapter 5, “Communications Performance” on page 79, for related information.

Workload Description and Data Interpretation:

- **Mail**

  The mail workload scenario was driven by an automated environment which executed a script similar to the mail workload from Lotus NotesBench. Lotus NotesBench is a collection of benchmarks, or workloads, for evaluating the performance of Notes servers. The results shown here are not official NotesBench measurements or results. The numbers discussed here may not be used officially or publicly to compare to NotesBench results published for other Notes server environments. For official AS/400 audited NotesBench results, see **www.notesbench.org**.

  Each user completes the following actions an average of every 15 minutes:

  - Open mail database
  - Open the current view
  - Open 5 documents in the mail file
  - Categorize 2 of the documents
  - Compose 2 new mail memos/replies
  - Mark several documents for deletion
  - Delete documents marked for deletion
  - Close the view

Mail Serving Performance Measurements: The following tables provide a summary of the measured performance data. These charts should be used in conjunction with the rest of the information in this section for correct interpretation. Results listed here do not represent any particular customer environment. Actual performance may vary significantly from what is provided here.

Table 54. Mail serving performance data.

Mail serving with Domino 4.6 (4.6.2 for 170-2292) AS/400 Server Models							
Model	**#Active Notes Users**	**#Domino Partitions**	**Main Storage**	**Response Time (seconds)**	**CPU %Busy**	**# Disk Arms**	**Disk %Busy**
170-2292 Mirrored	1350	1	1GB	.270	77	3	36
170-2292 RAID-5	1300	1	1GB	.172	71	4	N/A
S40-2207	17,600	16	40GB	.126	74	96	3
S40-2261	10,400	12	20GB	.050	57	47	12
S40-2261	10,400	12	12GB	.089	57	47	13
150-2269	30	1	64MB	.349	25	1	15

Note:

- Data shown above should not be compared to audited Notesbench results.
- Results may differ significantly from those listed here.
- These measurements are not meant to be interpreted as maximum user data points.

Based on the above measurements, the following are estimates of the 70% capacity for the most recently announced processors.

Table 55. Estimated Simple Mail Users

Mail serving with Domino 4.6.2 at 70% cpu
AS/400 Server and Traditional Models (* = V4R3)

Server Model/Feature	Traditional Model/Feature	n-way	Simple Mail Users at 70% cpu
S40-2208*	650-2189	12w	23900
S40-2207*	650-2188	8w	16500
S40-2261	650-2243	12w	12500
S40-2256	650-2240	8w	8700
S30-2260	650-2240	8w	8700
S30-2259	640-2239	4w	4600
S30-2258	640-2238	2w	2400
S30-2257	640-2237	uni	1400
S20-2166	NA	4w	3400
S20-2165	620-2182	2w	2000
S20-2163	620-2181	uni	900
S20-2161	620-2180	uni	500
NA	620-2179	uni	370
S10-2119	600-2136	uni	350
S10-2118	600-2135	uni	200
NA	600-2134	uni	130
NA	600-2129	uni	100
170-2386*	NA	uni	2300
170-2385*	NA	uni	2300
170-2292*	NA	uni	1300
170-2291*	NA	uni	600
170-2290*	NA	uni	400
170-2183	NA	uni	1300
170-2176	NA	uni	1300
170-2164	NA	uni	900
170-2160	NA	uni	500
170-2159	NA	uni	350

Note:

- Data shown above should not be compared to audited Notesbench results.
- Results may differ significantly from those listed here.
- These estimates are not meant to be interpreted as maximum user data points.

Mail Serving Performance Conclusions/Recommendations:

1. Domino 4.6 and 4.6.2 running on AS/400 V4R2 provides an industry leading scalable mail server. The AS/400′s unique subsystem architecture allows multiple Domino partitions on a single piece of hardware. Not only does this improve performance, but it allows for better manageability of groups of users.

2. The 170-2292 mirrored system with 1GB of main storage had very good response time at only 77% CPU utilization. This system had 1 drive in ASP 1

for OS/400, and 2 mirrored drives in ASP 2 for Notes data (mailboxes, name and address book, etc.)

3. The 170-2292 RAID-5 system with 1GB of main storage also had very good response time but could only support 1300 users with the 4 drives protected with RAID-5. The 3 additional I/Os for each write due to RAID-5 protection limited the number of users this configuration could support. Configuring additional arms would have relieved this bottleneck.

4. The S40-2207 with 40GB of main storage demonstrates how easily Domino for AS/400 scales upwards to large numbers of users and many partitions. Even at close to 18,000 users, the system still had available resources for more Domino users or other applications. Unfortunately, at the time of this writing, the S40-2208 12-way measurement results are not available.

5. The S40-2261 with 20GB of main storage ran at only 57% CPU utilization, and paging was very low. This means valuable main storage and CPU are available for other users and/or mission critical applications.

6. The S40-2261 main storage was reduced to 12GB with minimal impact on response time, CPU utilization, or disk utilization. The faulting rate was still very acceptable.

7. The 150-2269 also has leftover CPU and main storage capacity when running 30 active users. Even the single disk arm on this system was not overly utilized at 25%.

Domino Performance Tips/Techniques:

1. As the number of active users in a partition increases, contention for resources will cause an increase in CPU consumption for each user. Best results were achieved when the number of users was limited to 1000, even though the 170-2292 successfully ran 1350 users in a single partition. This was done to reduce the total cost of the system (Domino charges extra for Advanced Services which multiple partitions require).

2. Initial user connection places the heaviest load on the AS/400, requiring the largest amount of CPU, main storage, and disk resources. When sizing a system, ensure there is sufficient capacity for this activity as well as the typical peaks of activity throughout the day.

3. For the system storage pool in which the Domino server and users run, you will need to configure at minimum 7MB + 600KB per active user for small systems and 8MB + 800KB per active user for large systems. Note this does not include storage required for other system storage pools (e.g. *MACHINE, *SPOOL, etc.). Adding more main storage that what is recommended here will provide even better response times and will also provide capacity for future growth or workload peaks.

   Follow the faulting threshold guidelines suggested in the Work Management guide by observing/adjusting the memory in both the machine pool and the pool that the mail servers run in.

4. Disk space requirements:

   Typically, 30-50MB for each registered user′s mail database is recommended for planning purposes. For small configurations, you should also consider disk space necessary for Domino Server code and the name and address book.

5. AS/400 notes.ini parameter settings:

- For the first server job created, additional threads will be created as needed until 100 is reached. Then the secondary server job will be created with 100 threads all at one time. If this number for secondary threads is too high, you may either notice a general slowdown on the system as th entire set of threads is created, or you may get server errors if there are more than 2048 files open in a single job. Depending on your configuration, these errors begin to show up if the the number of threads per job approaches or exceeds 400. To set the initial and secondary threads to 100, put the following lines in the server′s NOTES.INI file.

  – SERVER_MAXINITIALTHREADS=100

  – SERVER_SECONDARYTHREADS=100

- MAILMAXTHREADS

  This is the maximum number of threads that the mail router job can create. The default is one thread per server port. Recommend increasing this number to improve mail- routing performance, which is especially important for mail hub servers.

- NSF_Buffer_Pool_Size

  This controls the size of the memory section used for buffering I/Os to and from disk storage. The best results for the 1350 user measurements were obtained when this was set to 507000000. If you make this too small and more storage is needed, Domino will begin using it′s own memory management code that adds unneeded overhead since OS/400 already is managing the virtual storage. Do not set this above 508000000, since doing so will cause Domino to calculate its own value, which may not be optimal.

6. AS/400 environment variable settings:

   - Notes_SHARED_POOLSIZE. Set to 12000000. Controls how Domino memory management is done. By increasing this value, you reduce the amount of management overhead. Do not make this variable larger than 12000000.

   - Notes_AS400_CONSOLE_ENTRIES set to to 10,000 (the default). This is the size of the console file that displays the status messages when you enter the DSPDOMCSL or WRKDOMCSL commands. As this file grows, the response time for these two commands increases.

   For more detail on the above settings, see the Domino Server Administrator Guide.

7. Use *LOOPBACK to prevent data from going out over network.

   For servers communicating on the same system, you can shortcut the communications path by using *LOOPBACK instead of transmitting the data out of the system just to have it come back again. See **http://www.as400.ibm.com/techstudio for details**.

8. Dedicate servers to a specific task

   This allows you to separate out groups of users. For example, you may want your mail delivered at a different priority than you want database accesses. This will reduce the contention between different types of users. Separate servers for different tasks are also recommended for high availability.

9. HTTP serving performance tips:

- Set MAXIMUM_ACTIVE_THREADS in names.nsf

  This is the limit of the number of active threads. When the limit is reached, the next browser request then has to wait for a free thread. The default is 40. Recommend changing this to 100 on large systems.

- Set MINIMUM_CACHE_SIZE in names.nsf

  This is the cache size used by Domino for the HTTP server threads for the conversion process from a Domino database to HTML. (This includes the conversion of text, .gifs, .jpgs) The default is 50. Recommend increasing to a larger size. Do not set to a value that exceeds the total system memory (RAM) size.

10. MIME format.

    For users accessing mail from both the Internet and Notes, store the messages in both Notes and MIME format. This offers the best performance during mail retrieve because a format conversion is not necessary. NOTE: This will take up extra disk space, so there is a trade-off of increased performance over disk space.

11. Full text indexes

    Avoid the use of full text indexes for mail files. You need to consider whether to allow users to create full text indexes for their mail files. These indexes are expensive to maintain since they take up CPU processing time and disk space.

12. Replication.

    To improve replication performance, you may need to do the following:

    - Use selective replication
    - Replicate more often so there are fewer updates per replication
    - Schedule replications at off-peak hours
    - Set up replication groups based on replication priority. Set the replication priority to high, medium, or low to replicate databases of different priorities at different times.

Domino Subsystem Tuning

The objects needed for making subsystem changes to Domino are located in library QUSRNOTES and have the same name as the subsystem that the Domino servers run in. The objects you can change are:

- Class (timeslice, priority, etc.)
- Subsystem description (pool configuration)
- Job queue (max active)
- Job description

The system supplied defaults for these objects should enable Domino to run with optimal performance. However, if you want to ensure a specific server has better response time than another server, you could configure that server in its own partition and change the priority for that subsystem (change the class), and could also run that server in its own private pool (change the subsystem description).

Mail Serving Capacity Planning

For more information on how to estimate system sizes for Domino server systems, see:

- **http://www.software.ibm.com/is/ibm-lotus/capplan.htm**
- **http://iws.as400.ibm.com/lotus_notes/notes.htm** under the section ′Domino for AS/400 Buying/Selling Guide′.
- **http://www.as400.ibm.com/tstudio/domino/index.htm** For an automated version of the above sizing guides.

Chapter 12. OS/400 Integration of Lotus Notes Performance

This section includes measurement data using the 6616 Integrated PC Server - on both the V3R2 CISC platform as well as the V3R7 Enhanced RISC platform. To use the AS/400 CPU data that is provided in this section to estimate utilization for other AS/400 models, please refer to Chapter 2, "AS/400 System Capacities and CPW" on page 3 for information on relative performance of AS/400 models. Due to differences in IPCS support, one should use CISC measurement values to estimate/project to other CISC models. Likewise, RISC values should be used to estimate other RISC models.

The V4R1 PRPQ (P84304) - IPCS EFS (Enhanced File System) further improves the performance capability of the IPCS - particularly those with more than 64MB of memory. It is able to manage/use large amounts of IPCS memory(RAM) to provide efficient file/data caching service to OS/2 and applications such as Notes. EFS is available for all models of the IPCS including those used in V3R2, V3R7, and V4R1 configurations. (**Note** A primary component of EFS is a version of OS/2 HPFS386.)

Performance information for OS/400 Integration of Lotus Notes(**) is included in this section. The information will be divided into two general areas: number of Lotus Notes clients that can be supported, and performance guidelines for DB2 Integration function.

For a complete overview and understanding of OS/400 Integration of Lotus Notes, please refer to the following resources:

- OS/400 Integration of Lotus Notes (SC41-3431-02) V3R7Enh
- OS/400 Integration of Lotus Notes (SC41-3431-01) V3R2
- Using Lotus Notes on the Integrated PC Server (IPCS) for AS/400 (SG24-4779-00). This is a Redbook publication.

The File Serving IOP (FSIOP) has been renamed the Integrated PC Server (IPCS). These terms are used interchangeably in this section and refer to the following models:

- IPCS Type 6506 (based on i486 66mHz CPU; RAM size up to 64MB)
  - 6506 was previously released and is compatible with AS/400 RISC and CISC models (except Model 150)
- IPCS Type 2850 (based on Pentium 133mHz CPU; RAM size up to 128MB)
  - 2850 is available only on AS/400 Models 150
- IPCS Type 6616 (based on Pentium 166mHz CPU; RAM size up to 256MB)
  - 6616 is compatible with AS/400 RISC and CISC models (except Model 150)

OVERVIEW: Substantial improvements are seen on the IPCS type 6616 as compared to the 6506:

- 480 Mail users versus 200 (non-EFS)
- 700 Mail users (6616 with EFS)
- DB2 Integration Import rates (non-EFS)**
  - 2MB/minute vs .5MB/minute (2.1KB recordlength)
  - 2.8MB/minute vs 1MB/minute (6.2KB recordlength)
- DB2 Integration Shadowing rates (Inserts, Updates, Deletes) (non-EFS)**
  - 100 changes/minute vs 15 changes/minute

(a 10,000 document Notes database)
- 62 changes/minute vs 7.5 changes/minute
(a 20,000 document Notes database)

**Note: Measurement data is not available for these DB2 Integration benchmarks with EFS. No performance improvements are expected from EFS caching since all data would be ′new′. However there may be some improvements due to more efficient I/O handling by the EFS.

12.1 Number of Notes Clients Supported

The Notes server on an IPCS can support different numbers of users depending on the type and number of requests from the Notes clients. Three workload scenarios are described below and measurement results using these scenarios are shown in the tables following the descriptions.

Workload Scenario Descriptions:

- **Mail**

  The mail workload scenario was driven by an automated environment which executed a script similar to the mail workload from Lotus NotesBench. Lotus NotesBench is a collection of benchmarks, or workloads, for evaluating the performance of Notes servers. The results shown here are not official NotesBench measurements or results. The numbers discussed here may not be used officially or publicly to compare to NotesBench results published for other Notes server environments.

  Each user completes the following actions an average of every 15 minutes:

  - Open mail database
  - Open the current view
  - Open 5 documents in the mail file
  - Categorize 2 of the documents
  - Compose 2 new mail memos/replies
  - Mark several documents for deletion
  - Delete documents marked for deletion
  - Close the view

- **Mail and Discussion**

  The Mail and Discussion workload scenario was driven by the same (as Mail) automated environment described above. It executes a script containing representative scenarios/sequences similar to Lotus NotesBench′s. Again, these numbers are not to be used officially or publicly to compare to NotesBench results published for other Notes server environments.

  Each user completes the following actions an average of every 15 minutes:

  - Open mail database
  - Make sure 50 documents exist
  - Open the current view
  - Open 5 documents in the mail file
  - Categorize 2 of the documents
  - Compose 2 new mail memos/replies
  - Mark several documents for deletion

- Delete documents marked for deletion
- Open a discussion database
- Make sure 200 documents exist
- Open the current view
- Page down the view 2 times
- Set the unread list to a randomly selected 30 documents
- Open the next 3 unread documents
- Close the view

- **Mail and Discussion with Import**

  For the mail and discussion with import workload scenario, three import requests (importing data from DB2/400 to a Notes database) were performed during the measurement. The imports executed sequentially and an import was active during the entire measurement. Please see 12.2, "Lotus Notes DB2 Integration Performance" on page 213 for additional information on the performance of import function. The mail and discussion portion of this workload scenario was the same as the mail and discussion workload scenario described above.

The tables below includes the data/measurement values that were published in the prior release of this document. They provide performance information and guidelines using the three workload scenarios described above:

IPCS Memory Guide for Mail Workload

Table 56. Memory guidelines for Mail Workload

Memory guidelines for number of Notes Mail users supported on Integrated PC Servers	
Memory Size	**Maximum Number of Users**
32MB	100
48MB	150
64MB	200
128MB	400 (w/EFS)
256MB	700 (w/EFS)

Observations: Memory requirements by the Mail workload is proportional to the number of users regardless of the IPCS′s processing speed. EFS is recommended for IPCSs with greater than 64MB. For 64MB or less, the Mail workload results indicate no performance gains.

A key resource in this environment is the internal cache for database (.NSF file) activity. This buffer pool size is specified with the NSF_BUFFER_POOL parameter in the NOTES.INI file which becomes effective at the startup of the Notes server application (STRNWSAPP). To monitor the actual use of this cache, use the Show Stat Database command at a Remote Console.

Here′s a sample response from Show Stat Database:

```
COMMAND SENT: show stat database
  Database.BufferControlPool.Peak = 784872
  Database.BufferControlPool.Used = 659592
  Database.BufferPool.Maximum = 264241152    <-- from notes.ini
  Database.BufferPool.Peak = 33926400        <-- peak use so far
  Database.BufferPool.PerCentReadsInBuffer = 97
  Database.BufferPool.Reads = 52172
  Database.BufferPool.Used = 33760468        <-- current use
  Database.BufferPool.Writes = 80848
  Database.NIFPool.Peak = 1308120
  Database.NIFPool.Used = 823933
  Database.NSFPool.Peak = 1242714
  Database.NSFPool.Used = 797506
```

The value shown for Database.BufferPool.Maximum is from default value specified in NOTES.INI or the available physical memory - whichever is smaller. We recommend that the Database.BufferPool.Used & .Peak be monitored regularly. ′.Maximum′, should be increased when ′.Peak′ approaches its value.

The Operating System for the IPCS is OS/2 Warp Connect with the HPFS dasd file system. A file system provides varying degrees of caching, concurrent open files, files sharing and file locks. HPFS in our case provides for a 2MB maximum disk cache - a rather modest size. EFS will greatly improve the HPFS dasd file system performance by expanding the cache size along with improved caching algorithms. I/O rates, a key indicator of caching effectiveness, has been reduced to about half. In the IPCS environment, the AS/400 CPU is used mostly to service dasd I/O requests. Thus when I/O rates are reduced, the AS/400 CPU utilization is reduced correspondingly. There is also significant reduction of the IPCS′s CPU utilization.

Suggested Technical References

An important share of this OS/400 product is based on IBM Austin′s work on Notes for the OS/2 PC platform. As such it may be instructive to study some of their technical papers regarding their experience and recommendations on the product set. These papers cover the ′Entry′ (using Warp Connect with HPFS) and ′High End′ (using Warp Server Advanced with HPFS386). Our IPCS product would correlate to the ′Entry′. Be aware of these differences when reading and/or applying the information provided.

IBM Austin′s papers may be accessed at these URLs:

- www.austin.ibm.com/pspinfo/noteonws.html#contents
- www.austin.ibm.com/pspinfo/noteonws1.html#flex
- www.austin.ibm.com/pspinfo/noteonws3.html#usage

At the third URL, there is background information and also recommendations on ′Tuning Notes Server Memory Usage′.

Please check the following Web pages for additional technical information, support and update notifications.

- www.as400.ibm.com/products/software/notes/performance.htm
- www.as400.ibm.com/notes/performance

AS/400 Machine Pool Guidelines

The following guidelines can be used to estimate the amount of AS/400 memory that will be required in the machine pool (System Pool #1) to support the indicated Integrated PC Server:

- IPCS 6616 - 1.4MB plus 1.8MB per port
- IPCS 6506 - 1.4MB plus 1.8MB per port
- IPCS 2850 - 4.5MB

IPCS 6506 and 6616 on CISC

Table 57. Users on Integrated PC Server running Notes

6506 on CISC (non-EFS)
Lotus Notes on IPCS-6506 64MB RAM
Various AS/400 models used, V3R2

Workload Scenario	**Number of Users**	**AS/400 Model(s) CPU(%)**	**AS/400 IO/Sec**	**Int PC Srvr(6506) CPU(%)**
Mail	100	5% 20S-2010	38	33
Mail	200	9% 20S-2010	68	66
Mail + Discussion	60	4% E25	18	33
Mail + Discussion with Import	60	13% E25	70	70

Note:
Average utilization of 16Mbps TRLAN was <1% for all scenarios

Table 58. Users on Integrated PC Server running Notes

6506 on CISC (non-EFS)
Lotus Notes on IPCS type 6506 w/64MB RAM
AS/400 model E25 used, V3R2, 13 DASD Arms, 64MB

Workload Scenario	**Number of Users**	**AS/400-E25 CPU(%)**	**AS/400-E25 IO/Sec**	**Int PC Srvr(6506) CPU(%)**
Mail	100	7	32	30
Mail	200	17	66	57

Note:
- Average utilization of 16Mbps TRLAN was <1% for all scenarios
- Virtual DASD (VDASD) support for IPCS on CISC is enhanced with multiple Storage Management tasks (3 for 6506; 7 for 6616)

Table 59. Users on Integrated PC Server running Notes

6616 on CISC (non-EFS) **Lotus Notes on IPCS type 6616 w/256MB RAM** **AS/400 model E25 used, V3R2, 13 DASD Arms, 64MB**				
Workload Scenario	**Number of Users**	**AS/400-E25 CPU(%)**	**AS/400-E25 IO/Sec**	**Int PC Srvr(6616) CPU(%)**
Mail	100	9	38	8
Mail	200	17	67	15
Mail	300	25	104	24
Mail	400	34	142	36
Note: - Average utilization of 16Mbps TRLAN was <1% for all scenarios - Virtual DASD (VDASD) support for IPCS on CISC is enhanced with multiple Storage Management tasks (3 for 6506; 7 for 6616)				

Observations: Table 57 on page 205 , Table 58 on page 205 and Table 59 summarize data for 3 CISC combinations of 6506 and 6616.

- IPCS 6506 on CISC
- IPCS 6506 on CISC
- IPCS 6616 on CISC

Note that the 6616 configuration has substantially extended the maximum number of users supported from 200 to more than 400. Although not listed in Table 59 , up to 480 users have been observed with this configuration. The main factors for this improvement are:

- Pentium/166mHz vs i486/33mHz; this represents processing speed ratio of about 3.5 times. This is confirmed by the IPCS′s CPU%: e.g. on E25 at 200 users - 6506 runs at 57%; 6616 is at 15%.
- IPCS (with large RAM): Configurations with greater than 64MB should benefit greatly from installing EFS. At this time there is no direct measurement data (of IPCS on CISC with EFS) - but it is expected that the pivotal parameters (DASD I/O rates , IPCS CPU%) will parallel the improvements shown in Table 64 on page 212 and Table 65 on page 212 (IPCS on RISC). We expect that the E25/6616 configuration of Table 59 above will support Mail users up to the 700 level with the E25 CPU utilization in the 40% range and the IPCS CPU in the 50% range.
- VDASD performance on CISC was enhanced by the PTF# NF14527, (on Cum. tape# 7063) which supported and used multiple tasks (3 for the 6506; 7 for 6616). See ″Multiple VDASD Tasks for CISC″ section below.

The AS/400 server model 20S-2010 shown in Table 57 on page 205 has a 5.9 Relative Processor Rating (RPR) for non-interactive work. Using the data for the 100 user mail workload, we can predict that this same workload would require approximately 6.6% CPU on an AS/400 model E25 which has an RPR rating of 4.0. **Note** This estimate is now validated (as correct) with data shown in Table 58 on page 205. See Chapter 2, “AS/400 System Capacities and CPW” on page 3 for additional information on RPR values to enable CPU comparisons for other AS/400 models.

All of the AS/400 CPU required to support both the Notes server and DB2 Integration function runs in non-interactive mode on server models. Server models have a better price performance than traditional models for this environment.

**** Note **** that for a given workload, differences in the rates shown in the tables are due to workload and measurement variability. These differences should not affect their usefulness as input to this level of system performance analysis.

****Also **** these CISC values and rates should be used to project performance onto other CISC models. Likewise use RISC values to project for other RISC models.

Multiple VDASD Tasks for CISC

The VDASD path has been enhanced for the support of IPCS on CISC. In the original IPCS release, there was a single VDASD task to perform all the DASD requests from the IPCS. With the multiple task enhancement, each 6506 will have three(3) of these tasks initiated at VARY ON time. Similarly, each 6616 will have seven(7).

Discussion: The service time of a vdasd task is basically that of the DASD service time - let′s say 15 milliseconds. This implies that a single task cannot perform more than about 60 to 70 I/Os per second in support of an IPCS. Response times would be extreme due to queuing times at such a max′d out server.

The limitation of the single VDASD task (original release) was masked by the 6506′s (relatively low) speed and memory capacity. With the 6616 whose speed and memory capacity is about tripled - a higher VDASD bandwidth was required. The multi-task VDASD support will perform well for the 6616 and for some workload environments, may also improve the performance characteristics of the 6506.

Performance Data The AS/400 Performance Tools can be used to create reports from data collected by the AS/400 system performance monitor (STRPFRMON). The Component report shows IPCS utilizations in the ″IOP Utilizations″ section, and shows job and task related data in the ″Job Workload Activity″ section.

The following section discusses the IPCS supporting tasks/names that will be reported and listed in the Performance Monitor reports.

Tasks related to IPCS processing

For CISC, the following names/types of tasks are used to process the DASD I/O requests for an IPCS:

- ROUTxx - handles communications across bus to IPCS
- #O00yy - DASD I/O server task to map HPFS space to AS/400 DASD space
- SM0nyy - storage management task to perform physical I/O to DASD

There will one of each of the first two task types, and multiples of the latter - where ′n′ will be 1, 2, and 3 for the 6506; and 1, 2, ... 7 for the 6616. There are additional tasks with similar names which are used for IPL and administrative processing.

For RISC, the following names/types of tasks are used to process the DASD I/O requests for an IPCS:

- FPHA-NWSDname - performs function similar to ROUTxx and #O00yy above
- SMDSTASKaa - multiple tasks (2 x # of DASD arms; with a maximum of 32 tasks) are used to process DASD I/O for all IPCSs on the system.

There will be a FPHA-NWSDname task for each IPCS on the system, and the SMDSTASKaa tasks are shared by all IPCSs on the system. Additional tasks, FPHI-NWSDname and FPN-NWSDname, also exist and are used for IPL and administrative processing.

For the scenario involving import, additional AS/400 jobs are used to retrieve data from DB2/400 for import to the Notes database. Jobs used for this function will have names similar to QZDASOINIT and QZRCSRVS.

Mail and Discussion with Import workload scenario

This scenario performed significantly more IO writes per second than the mail and discussion without import scenario. The high number of writes in this scenario occurred as a result of importing the data from DB2/400 and creating new documents in the Notes database. Having an import active used significantly more AS/400 and IPCS resources and impacted response times for the 60 attached users. See 12.2, “Lotus Notes DB2 Integration Performance” on page 213 for additional information and recommendations regarding the import function.

Multiple IPCS on CISC (non-EFS)

The following tabulation was provided in V3R2. Measurement data of this configuration/environment is not available for the 6616 IPCS. However, if required, one could project by scaling based on the preceding tables. For example if we were to replace the 6506s with 6616s, we would expect:

- each 6616 could support at least 400 users; total of 2400;
- each 6616′s CPU would be at about 36%
- I/Os would rise to 800+ per second; assuming sufficient DASD arms;
- CISC CPU would be at about 15%.
- Previous additional measurements with eight(8) 6506s achieved expected scalability.

Table 60. Multiple Integrated PC Servers running Notes

6x 6506s on CISC (non-EFS)
Integrated PC Servers running Lotus Notes on an AS/400 Model F90
V3R2
6 IPCS(6506) w/64MB RAM, each on separate 16Mbps LAN

Workload Scenario	Number of Users	AS/400 Model F90 CPU(%)	AS/400 Model F90 IO/Sec	Int PC Srvr(6506) CPU(%)
Mail	1200 (200 per 6506)	7%	404	59% 61% 65% 65% 65% 68%

Note:
Average utilization of each 16Mbps TRLAN was <1%
<This environment was not repeated with the IPCS Type 6616>

IPCS on RISC platforms.

The following sections cover results on the RISC platforms. Here are a couple of recommendations before the particulars:

- Use the (RISC) values in this section when projecting performance or capacity of other RISC models. This is warranted due to the differences of the VDASD path and structure on the RISC/SLIC as compared to the same on the CISC/VMC platform.
- Similarly, use the values from the Model 150 tables to project to other Model 150 models. An additional difference is that the Model 150′s 2850′s Pentium speed is 133mHz compared to the 6616′s 166mHz.

IPCS 2850 on RISC(Model 150)

Table 61. Users on Integrated PC Server running Notes

2850 on Model 150 (non-EFS)
Lotus Notes on IPCS type 2850 w/128MB RAM
AS/400 Model 150-2269 used, V3R7Enh, 4 DASD Arms; 96MB

Workload Scenario	Number of Users	AS/400 150-2269 CPU(%)	AS/400 150-2269 IO/Sec	Int PC Srvr(2850) CPU(%)
Mail	100	10	38	9
Mail	250	22	86	23
Mail	300	29	111	29
Mail	400	41	161	42
Mail + Discussion	50	7	25	8
Mail + Discussion	100	14	51	14

Note:
Average utilization of 16Mbps TRLAN was <1% for all scenarios
Measurement data on this configuration with EFS is not available.

Table 62. Users on Integrated PC Server running Notes

6616 on RISC (non-EFS) **Lotus Notes on IPCS type 6616 w/256MB RAM** **AS/400 model 400-2131 used, V3R7Enh, 12 DASD Arms; 96MB**				
Workload Scenario	**Number of Users**	**AS/400 400-2131 CPU(%)**	**AS/400 400-2131 IO/Sec**	**Int PC Srvr(6616) CPU(%)**
Mail	100	17	43*	8
Mail	200	26	67	15
Mail	300	43	115	24
Mail	400	57	156	36
Mail	490	69	191	54
Mail + Discussion	100	21	55	15
Mail + Discussion	150	31	80	21
Mail + Discussion	200	39	104	27
Mail + Discussion	250	49	131	35

Note:
Average utilization of 16Mbps TRLAN was <1% for all scenarios
*I/O rate is higher than expected(38)
Measurement data on this configuration with EFS is not available.

Observations

The I/O per second rate on the Model 150/RISC is up to 13% higher than on the CISC model (compared point for point). Part of the difference is the 5% lower I/O rates (per user) at the 200 and above, user load point on the CISC platform. Except for the RISC′s 100 user load point, the other points correlate fairly well. This consideration contributes to the rationale for using RISC data to project to other RISC and to using CISC data to project to CISC.

Table 63. Users on Integrated PC Server running Notes

6506 on RISC (non-EFS) **Lotus Notes on IPCS type 6506 w/64MB RAM** **AS/400 model 400-2131 used, V3R7Enh, 12 DASD Arms; 96MB**				
Workload Scenario	**Number of Users**	**AS/400 400-2131 CPU(%)**	**AS/400 400-2131 IO/Sec**	**Int PC Srvr(6506) CPU(%)**
Mail + Discussion	60	14	35	37
Mail + Discussion	100	20	52	57

Note:
Average utilization of 16Mbps TRLAN was <1% for all scenarios

Observations on the Mail and Discussion Workload

This workload generated a system load that is about 25% higher than the Mail-only load. This is based on the I/O rates listed above in Table 62 and Table 63.

Client and Server Configurations:

- AS/400 System and IPCS

  The AS/400 systems and IPCS used for these measurements were dedicated while executing the workload scenarios described above.
  Utilization data is provided in the tables for key system resources utilized during the measurements.

  Several combinations of IPCS and AS/400 systems were used and are explicitly specified in the headings of the measurement data tables:

  - 6506(64MB RAM) on AS/400 Model E25
  - 6616(256MB RAM) on AS/400 Model E25
  - 6506(64MB RAM) on AS/400 Model 400-2131
  - 6616(256MB RAM) on AS/400 Model 400-2131
  - 6616(256MB RAM) on AS/400 Model 40S-2112
  - 2850(128MB RAM) on AS/400 Model 150-2269
  - 6506(64MB RAM) on AS/400 Model 20S-2010 (w/100mHz clients)
  - Multiple 6506(64MB RAM) on AS/400 Model F90 (w/100mHz clients)

- Notes clients Except where noted, all the the PC/Clients used were OS/2 Notes Clients running on Pentium/133mHz hardware.

IPCS with EFS on RISC platforms

The following tables contain results from an IPCS with EFS on a AS/400 RISC model. Note that this is a relatively high speed server system (compared to the other models used in previous measurements). Default EFS cache sizes have been selected to optimize performance in the Notes environment.

For the Mail workload, use of EFS (IPCS with greater than 64MB of RAM):

- reduced DASD I/O request rates

  Compared to the previous (non-EFS) measurement data, EFS has lowered I/O rates by 40% to 60%.

- reduced AS/400 CPU consumption

  Since the AS/400 CPU is used mainly to service DASD I/O requests, its CPU utilization (or savings) will be proportional to the I/O rates.

- reduced IPCS CPU consumption

  IPCS CPU savings at the high end of the workload is about 40%; and no savings at the low end of the workload curve.

- increased capacity/throughput

  Capacity of a 256MB IPCS has increased to the 700 level from a previous high of 490 (reached in a non-EFS environment).

  Note that the capacity limits reached here are not due to excessive processor(s) and I/O utilizations; but is due to an OS/2 memory management limitation.

For the DB2 Integration environment and workloads, EFS may provide minimal performance improvement. It should however, provide equivalent performance to the non-EFS environment.

Following are data from runs on an IPCS with 256MB, 128MB and 64MB.

Table 64. Users on Integrated PC Server running Notes

6616 with EFS on RISC
Lotus Notes on IPCS type 6616 w/256MB RAM
EFS Cache size set at 25% (=64MB)
AS/400 model 40S-2112 used, V3R7Enh, 8 DASD Arms; 128MB

Workload Scenario	**Number of Users**	**AS/400 40S-2112 CPU(%)**	**AS/400 40S-2112 IO/Sec**	**Int PC Srvr(6616) CPU(%)**
Mail	100	1.1	22	9
Mail	200	1.6	32	13
Mail	300	2.8	56	20
Mail	400	4.0	79	25
Mail	500	5.0	100	34
Mail	600	6.0	121	40
Mail	700	8.5	165	49

Note:
Average utilization of 16Mbps TRLAN was <1% for all scenarios

Table 65. Users on Integrated PC Server running Notes

6616 with EFS on RISC
Lotus Notes on IPCS type 6616 w/128MB RAM
EFS Cache size set at 25% (=32MB)
AS/400 model 40S-2112 used, V3R7Enh, 8 DASD Arms; 128MB

Workload Scenario	**Number of Users**	**AS/400 40S-2112 CPU(%)**	**AS/400 40S-2112 IO/Sec**	**Int PC Srvr(6616) CPU(%)**
Mail	100	1.1	22	8
Mail	200	2.1	39	13
Mail	300	4.0	72	21
Mail	400	5.3	96	27

Note:
Average utilization of 16Mbps TRLAN was <1% for all scenarios

Table 66. Users on Integrated PC Server running Notes

6616 with EFS on RISC **Lotus Notes on IPCS type 6616 w/64 MB RAM** **EFS Cache size set at 12.5% (=8MB)** **AS/400 model 40S-2112 used, V3R7Enh, 8 DASD Arms; 128MB**				
Workload Scenario	**Number of Users**	**AS/400 40S-2112 CPU(%)**	**AS/400 40S-2112 IO/Sec**	**Int PC Srvr(6616) CPU(%)**
Mail	100	1.9	33	10
Mail	200	3.2	57	14

Note:
Average utilization of 16Mbps TRLAN was <1% for all scenarios

12.2 Lotus Notes DB2 Integration Performance

This section contains performance information for three functions provided by Lotus Notes DB2 Integration: (**Note** Currently, there are no measurement or experimental data available for DB2 Integration on IPCS with EFS.) Additional information is in the section “IPCS with EFS on RISC platforms” on page 211

- Import

  Import refers to the capability to create a Lotus Notes database from existing data stored in a DB2/400 database.

- Shadowing

  While performing an import, the shadowing option can be specified which will initiate automatic updates to the imported Lotus Notes database based on on-going changes occurring to the DB2/400 files from which the data was imported.

- Exit Program

  Exit program refers to the ability to create and register an exit program that periodically updates DB2/400 data based on changes made to a Lotus Notes database.

Importing DB2/400 data to a Lotus Notes database

In addition to the considerations and recommendations provided in this section, Lotus Notes recommendations for Notes databases should also be reviewed. For example, Lotus Notes recommends that Notes databases be less than 100MB in size. As Notes databases grow to this size and have large numbers of documents, some functions such as initially opening a Notes database can take quite a long time.

The following tables provide measurement data that was collected for imports with varying numbers of rows, sizes of rows, and number of fields per row. In the examples in this section, each imported row from DB2/400 becomes a Notes document. The data was imported from the same AS/400 system as where the IPCS Notes server resides. Very little activity from Notes clients was present during the import measurements described in this section and the AS/400 system was not dedicated but had low resource utilization from other work. The numbers provided in Table 67 on page 214 can be used as guidelines to help estimate time to import data into a Notes database from DB2/400.

Table 67. Importing DB2/400 data to a Notes database

IPCS 6506 on CISC (non-EFS)
Importing DB2/400 data to a Notes database on an IPCS-6506 64MB RAM
(background task AMgr set to be 0% active during imports)
AS/400 D60, V3R2, MTU on NWSD = 15400

Number of Imported Rows	Bytes Per Row	Columns Per Row	Import Time	Resulting .NSF file size
1000	2150	10	5 min	3 MB
2000	2150	10	6 min	5 MB
4000	2150	10	10 min	10 MB
16000	2150	10	95 min	41 MB
20000	2150	10	100 min	51 MB
24000	2150	10	N/A	N/A
32000	2150	10	175 min	88 MB
1000	6233	21	6 min	7 MB
2000	6233	21	13 min	15 MB
4000	6233	21	25 min	29 MB
16000	6233	21	105 min	117 MB
20000	6233	21	N/A	N/A
24000	6233	21	171 min	175 MB
(background task AMgr set to be 10% active during the following imports)				
16000	6233	21	128 min	117 MB
20000	6233	21	182 min	146 MB
24000	6233	21	244 min	175 MB

Table 68. Importing DB2/400 data to a Notes database

IPCS 6616 on RISC (non-EFS) **Import DB2/400 data to a Notes database on IPCS-6616 256MB RAM** **(background task AMgr set to be 0% active during imports)** **AS/400 400-2131, V3R7Enh, 96MB, 12 DASD Arms, MTU on NWSD = 15400**				
Number of Imported Rows	**Bytes Per Row**	**Columns Per Row**	**Import Time**	**Resulting .NSF file size ****
1000	2150	10	<2 min	2+ MB
2000	2150	10	2 min	4+ MB
4000	2150	10	4 min	8+ MB
16000	2150	10	16 min	33 MB
20000	2150	10	20 min	41 MB
24000	2150	10	23 min	49 MB
32000	2150	10	31 min	65 MB
1000	6233	21	3 min	7 MB
2000	6233	21	5 min	13 MB
4000	6233	21	9 min	27 MB
16000	6233	21	37 min	106 MB
20000	6233	21	46 min	132 MB
24000	6233	21	57 min	160 MB

Note:
Note Smaller resulting .NSF file size than in Table 67 Reasons unknown.

Observations

Great performance improvements have been achieved in the DB2/400 Import function with the 6616 IPCS!

- with 2150 bytes/row database, we get up to 6x response time improvement (16,000 imported rows)
- Import rate is about .4 MB/minute vs 2 MB/minute
- with 6233 bytes/row database, we getup to 3x response time improvement (24,000 imported rows)
- Import rate is about 1 MB/minute vs 2.8 MB/minute

Caution The Import function may consume substantial resources and could impact other foreground jobs! Note the utilizations and rates as follows during the Import of 16,000 rows (6233 bytes/row) to a Notes database.

- RISC CPU% at 46%
- I/O per second at 130
- IPCS CPU% at 25%

Recommendations

- Maximum Transfer Unit

  It is recommended to set the Maximum Transfer Unit (MTU) size for the internal LAN to 15400 in order to provide maximum data throughput when using the DB2 Integration functions. The MTU parameter is found on the Network Server Description (NWSD).

- AMgr settings

  The AMgr background task affects the performance of import function when it is active. It was observed that with default setting of 50% active daytime and 70% nighttime, that the time to import can vary greatly. These settings can be adjusted and it is recommended for optimum import performance that the setting for percent active time be set near 0 for the time of day when the imports will be occurring. It is also possible to not have this task start up at all by editing the NOTES.INI file and taking it out of the list of tasks that get started on the Notes server. Please refer to the Lotus Notes 4.0 Administrator′s Guide for detailed information. The AMgr (Agent Manager) task allows control of who can run agents and when they can run on each server. From the data in Table 67 on page 214 it can be seen that with the AMgr task 10% active that Import times were greater than when AMgr was set to 0% active.

- Starting an import

  Import requests are queued and processed one at a time in the order they are requested. If a combination of large and small imports are to be requested, it may be appropriate to submit requests for the smaller imports first (for example if Notes clients are waiting to use them), rather than delaying them the length of time of the larger imports. An agent program wakes up every 2 minutes on the Integrated PC Server to check for import requests, so import requests may take 2 minutes before they begin processing.

- Opening a database

  After importing a database the default view is to show all of the columns. Choosing fewer columns through a view can significantly improve the time to do the first open. Notes databases can takes minutes to open the first time if they contain tens of thousands of documents. Refer to Notes on-line help documentation for suggestions on improving view display times.

- Import impacts to Notes clients

  While an import is active, the performance of Notes clients can be impacted. When possible it is recommended to perform imports when the level of Notes client activity is low.

- Import versus Import with Shadowing consideration

  Importing a given number of rows will take somewhat longer if the import with shadow option is specified. Please refer to data in the next section on shadowing for additional information.

Shadowing DB2/400 data to a Lotus Notes database

When initiating an import, the user is given an option to start shadowing function for the DB2/400 database file(s) indicated in the import request. To shadow DB2/400 files to a Notes database, Data Propagator Relational/400 is used and the files to be imported from and shadowed must have journaling active. For further information on shadowing, please refer to the resources indicated at the beginning of this section as well as **DataPropagator Relational Capture and Apply/400 (SC41-3346)** and. Other parameters that can be specified when initiating an import with shadowing include the time of day and/or frequency at which the user desires the shadowing activity to occur. Considerations for these setting are discussed later in this section.

The data in the following tables provide examples of the time required to shadow various types of changes (inserts, deletes, updates) from a DB2/400 file to a Notes database.

Table 69. Shadowing DB2/400 data to Notes database

<table>
<tr><th colspan="2">IPCS 6506 on CISC (non-EFS)
Shadowing DB2/400 data to a Notes database on IPCS-6506 64MB RAM
(background task set to be 0% active during measurements)
Documents in Notes database were 2150 bytes
AS/400 D60 V3R2, MTU on NWSD = 15400</th></tr>
<tr><th>Description of shadowing changes made to 10,000 document Notes database</th><th>Time to Shadow Changes</th></tr>
<tr><td>200 Inserts</td><td>20 minutes</td></tr>
<tr><td>200 Deletes (spread throughout Notes DB)</td><td>12 minutes</td></tr>
<tr><td>200 Updates (spread throughout Notes DB)</td><td>11 minutes</td></tr>
<tr><td>100 Inserts, 100 Deletes, 200 Updates</td><td>27 minutes</td></tr>
<tr><th>Description of shadowing changes made to 20,000 document Notes database</th><th>Time to Shadow Changes</th></tr>
<tr><td>200 Inserts</td><td>49 minutes</td></tr>
<tr><td>100 Deletes (spread throughout Notes DB)</td><td>14 minutes</td></tr>
<tr><td>200 Updates (spread throughout Notes DB)</td><td>24 minutes</td></tr>
<tr><td>100 Inserts, 100 Deletes, 200 Updates</td><td>54 minutes</td></tr>
<tr><td colspan="2"><b>Note:</b>
The ″time to shadow changes″ only includes the time to shadow the changes to the Notes database, the DB2/400 file changes had already occurred</td></tr>
</table>

Table 70. Shadowing DB2/400 data to Notes database

IPCS 6616 on RISC (non-EFS) **Shadowing DB2/400 data to a Notes database on IPCS-6616 256MB RAM** **(background task set to be 0% active during measurements)** **Documents in Notes database were 2150 bytes** **AS/400 400-2131 V3R7Enh, 12 DASD Arms, 96MB; MTU on NWSD = 15400**	
Description of shadowing changes made to 10,000 document Notes database	**Time to Shadow Changes**
200 Inserts	3 minutes
200 Deletes (spread throughout Notes DB)	2 minutes
200 Updates (spread throughout Notes DB)	2 minutes
100 Inserts, 100 Deletes, 200 Updates	4 minutes
Description of shadowing changes made to 20,000 document Notes database	**Time to Shadow Changes**
200 Inserts	6 minutes
100 Deletes (spread throughout Notes DB)	2 minutes
200 Updates (spread throughout Notes DB)	3 minutes
100 Inserts, 100 Deletes, 200 Updates	7 minutes
Note: The ″time to shadow changes″ only includes the time to shadow the changes to the Notes database, the DB2/400 file changes had already occurred	

Observations

This set of Inserts, Deletes, Updates test cases also exhibited substantial improvements in response times averaging

- 6x improvement when operating on the 10,000 document Notes database.
- 8x improvement when operating on the 20,000 document Notes database.

Conclusions/Recommendations:

- Inserts

  Shadowing inserts to Notes databases occurs faster for smaller databases and proportionately longer for larger Notes databases. From the data in Table 69 on page 217

  – with 6506/CISC, inserts into the 10,000 document Notes database occurred at a rate of approximately 600 per hour, while inserts into the 20,000 document database occurred at a rate of approximately 250 per hour.

  – with 6616/RISC, it′d be 4000/hour and 2000 per hour respectively.

- Deletes and Updates

  From Table 69 on page 217

  – On the 6506/CISC platform:

    when performing Shadowing, Deletes and Updates on the 10,000 document (Notes database); the deletes and updates occurred at a rate of approximately 1000 per hour, and at a rate of approximately 400 to 500 per hour for the 20,000 document Notes database.

Similar to insert, the size of the Notes database will typically impact the rate at which shadowed deletes and updates can occur. For these examples the documents to be deleted or updated were spread throughout the Notes database. If the documents had been found at the top of the Notes databases in both cases, the rates for shadowing delete and update would have been similar for the 10,000 and 20,000 document databases.

  - on the 6616/RISC platform, the rates are 6000/hour and 3600/hour respectively.

- Shadowing vs Importing

  From the data in Table 69 on page 217 above, it can be seen that the various types of changes can be shadowed at the rate of hundreds of changes per hour. The entire 10,000 document database that was being updated in the first part of Table 69 on page 217 was imported in 32 minutes. Many issues need to be considered regarding the use of the Notes database, but if the changed data is not required by applications or users in a real time manner, it may be a consideration to repeatedly import the entire database if thousands of changes are anticipated for the DB2/400 files that would be shadowed.

- Shadowing impact on Notes clients

  Like import, shadowing activity can significantly impact the performance of active Notes clients. When initiating the import with shadowing request, consider the settings for frequency of shadowing interval. This interval determines how often the Notes server checks to see whether shadowing changes are queued up and then begins making the changes to the Notes database. To avoid impact to Notes clients, attempt to schedule shadowing activity during times when client activity is low.

- Shadowing impact on Import

  Import with Shadowing option ON will take longer than Import without Shadowing. From data in Table 67 on page 214, a 20,000 row (2150 bytes per row) import took 100 minutes. Importing the same 20,000 rows with the shadowing option specified took 132 minutes (data for time to import with shadowing is not shown).

- Estimating shadowing rates

  Data from Table 69 on page 217 can be used to estimate time to complete various combinations of shadowing activity. Using the individual rates for inserts, updates, and deletes of a 10,000 document database to estimate the time to perform 100 inserts, 100 deletes, and 200 updates would yield an estimate of 27 minutes. This is in fact how long it actually took when that specific combination of changes was measured as indicated in the table. Data in Table 70 on page 218 can be used in a similar manner.

Exit Program: data from a Notes database to DB2/400

It is up to the user to create and register the Exit program to process the data once it is stored in the AS/400. The data in Table 71 provides examples for times it took to send the changes from the Notes database to DB2/400 for update. This environment used a 10,000 document Notes database and the rates of changes for inserts, updates and deletes are very similar to those shown in Table 69 on page 217 for Shadowing changes.

Table 71. Exit Program function

IPCS 6506 on CISC (non-EFS) **Exit Program: changed data from Notes database to DB2/400, does not include processing to apply changes to DB2/400** **(background task set to be 0% active during measurements)** **Documents in Notes database were 2150 bytes** **AS/400 D60 V3R2, IPCS-6506 64MB RAM, MTU on NWSD = 15400** **<This environment was not repeated with the IPCS type 6616 or 2850>**	
Description of changes sent to DB2/400 based on changes to Notes database	**Time to Shadow Changes**
200 Inserts	22 minutes
200 Deletes (spread throughout Notes DB)	12 minutes
200 Updates (spread throughout Notes DB)	13 minutes
100 Inserts, 100 Deletes, 200 Updates	29 minutes

Chapter 13. Language Performance

From its inception, AS/400 was designed to easily incorporate new technology while protecting customers′ substantial investments in software. AS/400 Advanced Application Architecture effectively isolates applications from low level changes in the system. Changes to accommodate new technology are made below the Technology Independent Machine Interface, which is preserved across releases. As a result, applications running above the Technology Independent Machine Interface do not have to change.

The program creation process on AS/400 is one of the key elements in isolating applications from low level changes in the system. With RISC there have been changes to this support that directly affect compile times. To discuss these changes and the significant improvements in ILE compile times, this section will cover the following topics:

- How programs are created, and what changes for RISC.
- Relative performance comparisons of ILE and OPM compile time on RISC versus CISC.
- Runtime performance
- Program object size comparisons
- Working memory guidelines for compiles

CISC Program Model

When you first compile an AS/400 program, the system goes through a two step process (see Figure 33 on page 222). The system first creates a program creation template (PCT). The PCT is independent of the instruction set that is used by the processor. The system then translates the PCT into a set of machine instructions. The end result is an executable program (*PGM) or an ILE module (*MODULE). Under OS/400 V2R3 or later CISC releases, two translators are available, one for OPM languages and one for ILE languages.

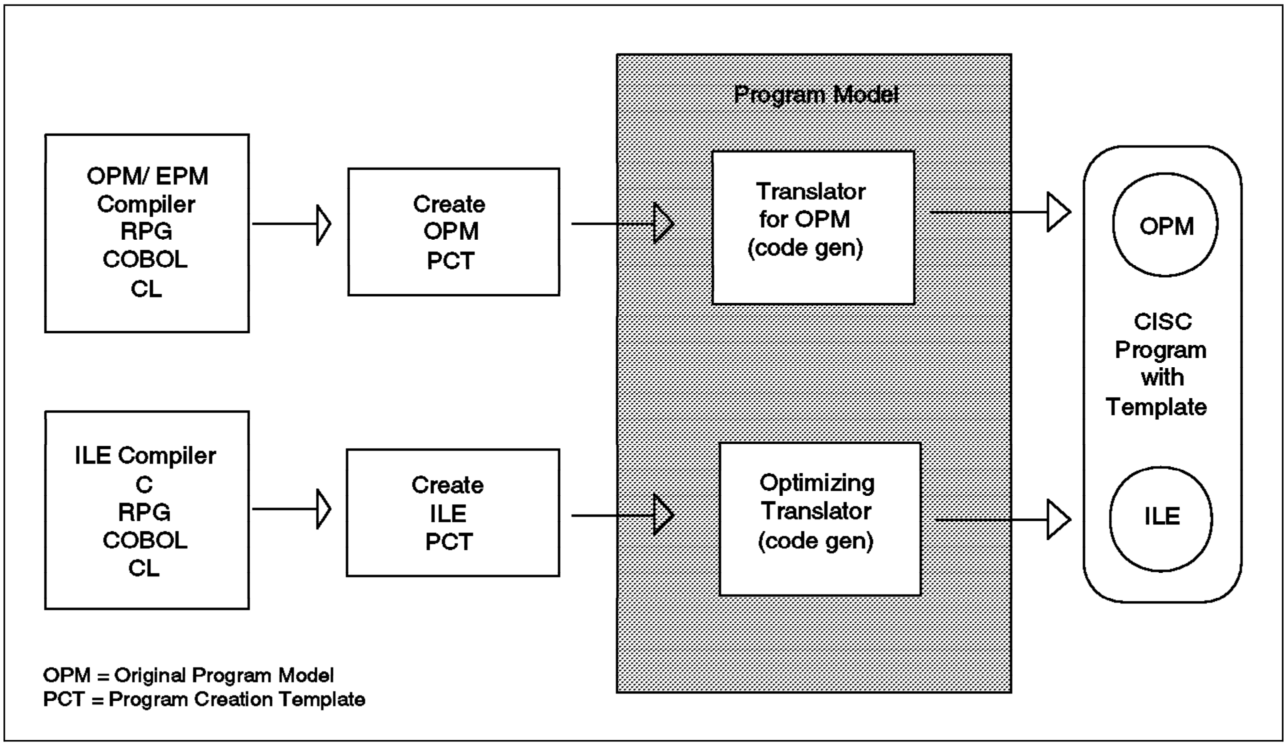


Figure 33. Program Model on CISC

The PCT is stored with the program unless you remove the program′s observability. The AS/400 Advanced Application Architecture allows you to translate the same PCT into RISC instructions or CISC instructions. With this capability, you can move programs between RISC and CISC platforms without recompiling. As discussed earlier, this is a major advantage of the AS/400 as it allows you to incorporate the latest technology without having to rewrite or recompile your application.

Program Model on RISC

To get the full performance benefit of the RISC processor architecture, an advanced code generation technology is required. Code optimization and code scheduling increases your application performance by eliminating redundant instructions and reducing unused cycles. This advanced technology is part of the Optimizing Translator.

To make this technology available to the existing OPM compilers, the OPM PCTs are automatically transformed into ILE PCTs. These ILE PCTs are then used by the Optimizing Translator to generate optimized RISC instructions (Figure 34 on page 223). With this you get the benefits of advanced code generation and application investment protection automatically. The additional conversion step does affect the compile time performance of OPM programs which is discussed in the next section.

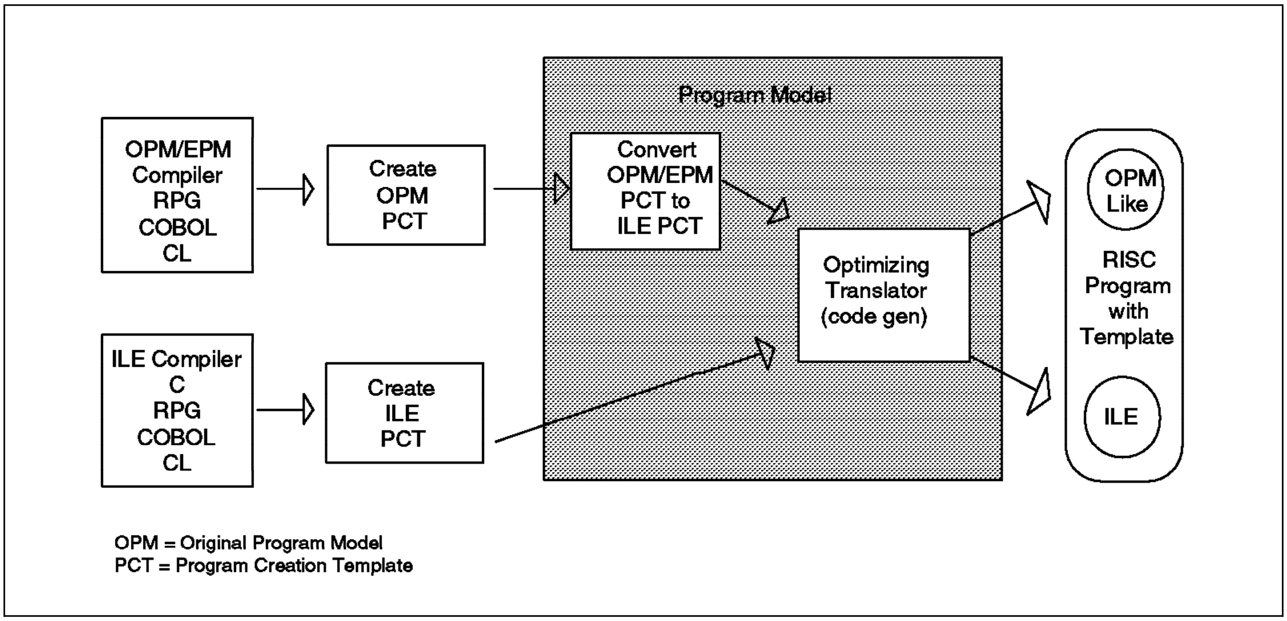

Program Model
OPM/EPM Compiler RPG COBOL CL
Create OPM PCT
Convert OPM/EPM PCT to ILE PCT
Optimizing Translator (code gen)
ILE Compiler C RPG COBOL CL
Create ILE PCT
OPM Like
RISC Program with Template
ILE
OPM = Original Program Model
PCT = Program Creation Template

Figure 34. Program Model on RISC

13.1 Compile Time Performance

The purpose of this section is to provide you with general information on compile time performance when moving to RISC. First, here′s a list of changes in RISC that significantly impact compile time. The list begins with the two factors mentioned in the preceding section:

- RISC code gen
- OPM program template conversion
- Memory requirements
- Changes to ILE *BASIC optimizations
- The new ILE/C optimization level of 40.

The tables below depict the relative performance of OPM and ILE compile time on V3R6 and V3R7 versus V3R1 for both RPG and COBOL. The V3R1 OPM compile time is used as a base (equal to 1.0). This information can not be used to compare RPG to COBOL compile time performance. Compile times for CISC have not changed greatly since V3R1 so those benchmarks are not being re-run.

The relative performance numbers included in the tables are averages based on results in a controlled environment, where compiles were done at the lowest optimization level and sufficient memory was available to keep paging to a minimum. Depending on your compile workload, your results may vary.

*Table 72. V3R6 Versus V3R1 Compile Time, Optimization *NONE.*

Program Type	**RPG and COBOL**	
	V3R1	**V3R6 and V3R7(*)**
OPM	1.0	1.2-1.7
ILE	greater than 2.0	1.0

Note:

A larger value signifies longer compile time.

(*) For the particular set of programs we used in this measurement, the compile times for V3R6 and V3R7 came out equal. However, there are programs that may realize a compile time improvement with V3R7 and some that may degrade. See "V3R7 Compile Time Conclusions" on page 225 and 13.4, "Working Memory Guidelines for Compiles" on page 231 for more information.

*Table 73. V3R6 Versus V3R1 Compile Time, Optimization *FULL*

Program Type	**RPG and COBOL**	
	V3R1	**V3R6 and V3R7(*)**
OPM *OPTIMIZE	1.0	4.4
ILE *FULL	4	3

Note:

A larger value signifies longer compile time.

(*) For the particular set of programs we used in this measurement, the compile times for V3R6 and V3R7 came out equal. However, there are programs that may realize a compile time improvement with V3R7 and some that may degrade. See "V3R7 Compile Time Conclusions" on page 225 and 13.4, "Working Memory Guidelines for Compiles" on page 231 for more information.

The following table shows the relative performance of ILE C compile time on RISC versus V3R1. The V3R1 ILE C compile time is used a base (equal to 1.0). As stated above, the results were obtained in a controlled environment and your results may vary.

*Table 74. RISC Versus V3R1 ILE C Compile Time, Optimization *NONE.*

Program Type	**ILE C**	
	V3R1	**V3R6 and V3R7(*)**
ILE C	1.0	0.5-0.8

Note: A larger value signifies longer compile time.

A larger value signifies longer compile time.

(*) For the particular set of programs we used in this measurement, the compile times for V3R6 and V3R7 came out equal. However, there are programs that may realize a compile time improvement with V3R7 and some that may degrade. See "V3R7 Compile Time Conclusions" on page 225 and 13.4, "Working Memory Guidelines for Compiles" on page 231 for more information.

V3R6 Compile Time Conclusions

The compile process requires more memory for RISC than CISC. The following conclusions are based on using sufficient memory to keep paging to a minimum. Compile times are also sensitive to the optimization level.

- The compile time of both ILE RPG and ILE COBOL have improved significantly on RISC. In both cases, RISC ILE compiles are in general up to 2 times faster than on V3R1 when compiling with optimization level (*NONE).
- V3R6 ILE RPG and COBOL compile times for optimization level (*NONE) are on the average equivalent to OPM compile times on V3R1, but can range from about 20% slower up to 20% faster.
- ILE C compile time on V3R6 has improved significantly over V3R1 - on the average from 20% up to 2 times faster. The most improvement will be in large programs at optimization level (*NONE). The least improvement will be in small programs at higher optimization levels.
- V3R6 OPM compiles are approximately 20-70% slower than on V3R1 for non-optimized programs. The additional time is the result of the automatic translation to allow OPM languages to utilize the Optimizing translator.
- Optimized ILE RPG and COBOL compiles have improved 30-40%. OPM optimized compiles are much longer than on V3R1. This is due to the more difficult optimizations attempted by the V3R6 optimizing translator when compared with the V3R1 OPM optimizing translator.

V3R7 Compile Time Conclusions

- V3R7 compile times on average have improved 5% to 25% over V3R6. However, some environments will see much greater increases, and some will experience degradations. Compiles being done in memory-constrained environments may show considerably larger improvement due to reductions in compiler working set size. As an example, see the data for a 5300 MI instruction OPM RPG program in Table 76 on page 232. Data points in the recommended 16 Mbyte to 20 Mbyte pool size range are showing a 5X to 6X improvement for this particular program. However, in an unconstrained environment this program shows a more modest 30% improvement.
- Degradations may be experienced in large compiles (modules with procedures larger than 500k bytes) at OPT(*FULL) or above. These compiles may take twice as long as they did in V3R6. This is caused by a change in register optimization strategy. Significantly better runtime code is produced at the cost of this compile time. If the compile time is too long, the module should be restructured with smaller procedures or the compiler optimization level reduced to OPT(*BASIC).

Compile Time Recommendations

- When possible, application developers should move to ILE. In addition to improved compile times, ILE offers many advantages over OPM, such as modularity, static binding, common run-time services, and improved code optimization.

  ILE RPG is shipped with a command, CVTRPGSRC, which can be used to migrate your RPG III source code to RPG IV. The *ILE RPG/400 Programmers Guide, Appendix B*, contains a detailed description of the conversion process, with examples to help you identify and quickly resolve potential conversion problems. Another source for information on the conversion process is the Redbook entitled *Moving to ILE RPG (GG24-4358)*.

For conversion and compatibility considerations between OPM COBOL and ILE COBOL for OS/400, please refer to the *ILE COBOL/400 Programmers Guide, Appendix G*.

- The following suggestions help in managing and improving compile times:
  - For initial compiles use OPTION(*NOGEN), and optimization *NONE or *NOOPTIMIZE.

    *NOGEN compiles the module or program but does not create a program object. It can be used to fix and edit compile errors.

    Using optimization *NONE or *NOOPTIMIZE can dramatically reduce compile times. Optimized compiles can be expected to take at least 3-5 times longer than compiles at optimization *NONE or *NOOPTIMIZE. Once the application is debugged and ready for production use, compile it at the appropriate optimization level, and conduct a final test. Typically RPG and COBOL programs should be compiled at optimization level (*NONE) or (*BASIC), and C programs at optimization level (*FULL) or level 40.
  - Use the appropriate working memory size

    See Working Memory Size Guidelines section below.
  - Compile in batch rather than interactively.
  - The following recommendations hold for ILE applications:
    - Design modular applications

      Modular programming offers faster application development and a better ability to reuse code. Programs are developed in smaller, more self-contained procedures. These procedures are coded as separate functions, and then bound together to build an application. By building applications that combine smaller and less complex components, you can compile and maintain your programs faster and easier.
    - Use the value of DBGVIEW adequate for your purpose.

      Requesting debug information requires more compile time and creates larger objects. For example, DBGVIEW(*LIST) results in a slower compilation time than DBGVIEW(*STMT). If the level of debug information you need is that provided by DBGVIEW(*STMT), selecting *LIST would unnecessarily slow down compilation time and inflate object size.

13.2 Runtime Performance

Language runtime was not remeasured for V3R7. Some enhancements have been made to generated code, but overall these will not cause language performance to be significantly different from V3R6. Much more improvement will occur due to improvements in OS/400 functions used by applications.

RPG and COBOL Runtime Performance

Runtime measurements indicate that, in general, the performance of RPG and COBOL programs for both ILE and OPM on V3R6 is better than V3R1. Some applications will experience an overall improvement, up to 10%, while others may experience a slight degradation. The most improvement can be expected in applications that are doing numerical processing, such as financial analysis applications. In this case ILE programs realized up to a 40% improvement. The least improvement can be expected in applications that are I/O intensive. Applications with many exceptions may be slower.

C Runtime Performance

Overall, ILE C runtime performance has improved on V3R6.

The ILE C/400 runtime benchmarks are a set of programs of varying sizes which measure the runtime performance of typical C language applications.

The table below shows the relative improvement ratios for the C benchmarks at all optimization levels, when compared to V3R1. Each column compares runtime performance for benchmarks compiled at the same optimization level on both V3R6 and V3R1. Since optimization level 40 is new with V3R6, it is compared to V3R1 optimization level 30.

The optimization level numbers equate to:

OPT(10)	OPTIMIZE(*NONE)
OPT(20)	OPTIMIZE(*BASIC)
OPT(30)	OPTIMIZE(*FULL)
OPT(40)	new for V3R6

The numbers represent the average performance improvement ratio of programs running on V3R6 when compared to their performance on V3R1. Numbers greater than 1 indicate that the program runs faster on V3R6 when compared to a V3R1 system of the same Relative Performance Rating (RPR).

The benchmarks are grouped into categories to represent the type of application, or application function which they measure.

Table 75. V3R6 ILE/C runtime performance improvement compared to V3R1

Category	OPT(10)	OPT(20)	OPT(30)	OPT(40)
Utilities	2.5	2.6	2.9	3.1
Computational	2.2	2.5	2.2	2.2
String	1.1	1.1	1.3	1.2
Call Intensive	3.2	3.2	3.9	5.1
Stream I/O	2.4	2.4	2.6	2.6
Record I/O	1.5	1.5	1.5	1.5
All Benchmarks	2.1	2.2	2.2	2.3

The benefits of optimization for C are shown in Figure 35 on page 228. The measurements are ratios to optimization level (*NONE) or (10). OPT(10) is represented as the value 1.0 on the y-axis.

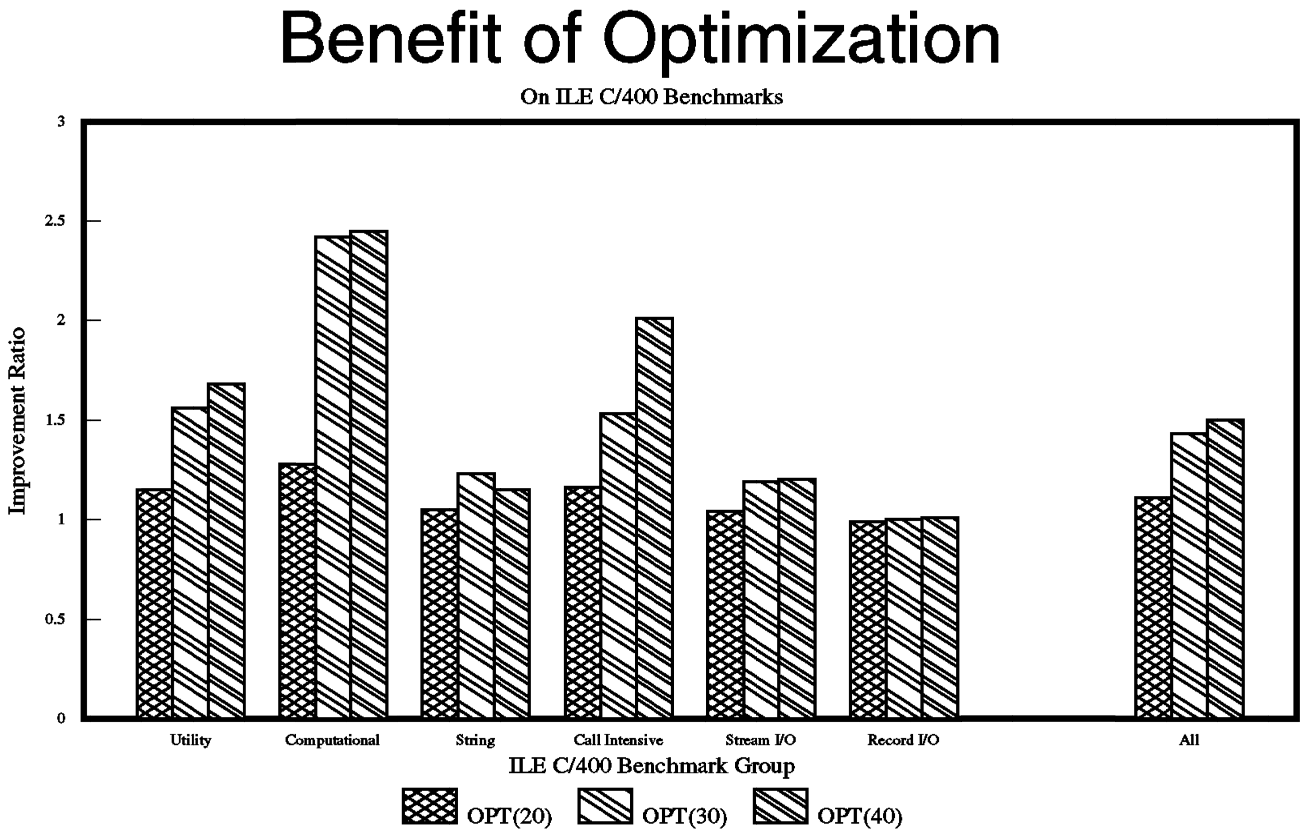


Figure 35. Comparison of ILE/C optimization benefit

Runtime Conclusions and Recommendations

Note: These conclusions are drawn from a set of measurements of runtime primitives made in the Rochester lab. They pertain to the actual benchmark programs. Other application programs may see varying performance benefits on V3R6 depending on structure and usage. These conclusions are meant as a general guide to understanding performance on V3R6, and are not meant to suggest minimum expected performance gains, or to guarantee performance of any particular application.

- String manipulation functions show the lowest gain when compared to V3R1. Applications manipulating longer strings may see relatively slower performance than on V3R1.
- Record I/O functions, which are faster than stream I/O, don′t show as much relative gain over V3R1 as do other C functions. This is because more of the CPU time is spent in OS/400 and database than for the other benchmarks.
- General logic and integer computation are much improved over V3R1; generally between 2X and 3X faster. Floating point applications gain less than integer based applications.
- Call intensive functions are much faster than on V3R1. At OPT(30) gains are about 3X. At OPT(40), performance of leaf functions (those functions not calling other functions), is particularly better, averaging about 5X faster.

  New with ILE C in V3R6 is direct mapping of stream I/O interface to the Integrated File System (POSIX) APIs. Optimum stream I/O performance is achieved when data is stored in the QOpenSys file system. Early measurements have indicated that stream I/O performance can improve significantly when using IFS.

Tradeoffs

- At OPT(10), average C applications gain about 2X relative to V3R1.
- OPT(20) yields about 10% performance improvement over OPT(10), at a cost of increased compile time (1.7X slower than OPT(10)).
- OPT(30) yields about 45% performance improvement over OPT(10), at a cost of increased compile time (4X slower than OPT(10)).
- OPT(40) yields about 5% performance improvement when compared to OPT(30). The compile time is about the same as OPT(30). This improvement depends on how many procedure calls are to leaf routines. Some programs may see a larger benefit from OPT(40).

13.3 Program Object Size Comparisons

Program Object Size Growth

On V3R6 there are several architectural factors which influence the size of program objects. First, the page size has increased from 512 bytes to 4K bytes (4KB). The larger page size is important in making storage management algorithms more efficient as the size of main storage continues to increase. The high end model 530 will support up to 4 gigabytes (4GB) of main storage, and this is likely to increase in future announcements. However, the 4KB page does impact the size of objects, particularly smaller objects, since objects must be aligned on 4KB boundaries rather than 512 byte boundaries.

Second, in general, the number of instructions for a comparable program is going to be larger on RISC than on CISC. This is referred to as code expansion. By the very nature of RISC design (efficient execution of simple instructions), it takes more instructions to do the same function as on CISC. For example, on RISC there are no storage-to-storage instructions; all data must be processed through registers. On CISC, moving data between two storage locations can be done with a single MVC (move character instruction). On RISC this requires a Load and then a Store instruction. In most cases, code expansion will have more of an impact on program object growth than the 4KB page size.

Conclusions and Recommendations

Due to the above factors, program object growth (for observable programs) when moving to V3R6 relative to V3R1 is as follows:

- ILE programs will grow on the average by 1-2 times.
- Observable OPM programs will grow on the average by 2-3 times.
- Non-observable OPM programs will grow on the average by 4 to 5 times. This range may be as high as 6 times for very small programs and as low as 3 times for very large programs.

The relative growth of non-observable OPM programs on V3R6 as compared to V3R1 will be greater than the relative growth for observable OPM programs because the majority of program growth is due to the code expansion of the executable part of a program object. The size of the program creation template (PCT) does not increase significantly, and for non-observable programs, the PCT has been removed. With the exception of observable programs with no compression, the ILE versions of programs are slightly smaller than the OPM versions. Also, the effectiveness of optimization as a size reduction tool is not large. A reasonable expectation would be that individual programs are reduced by one 4KB page per level of optimization.

To determine overall system DASD growth when moving to V3R6, you should use the Upgrade Assistant. For more information on the Upgrade Assistant refer to the manual *AS/400 Planning for PowerPC Technology, SA41-4154*.

Here are several options to consider to reduce storage requirements for your program objects:

- Compress the observable part of the program

  After you build your application, before it goes into production, you can reduce all unused parts of the object until they are actually needed by using the CPROBJ (Compress Object) command. Specify *OBS on the PGMOPT parameter. You should NOT remove program observability to reduce program size.

- For ILE programs, use modular design techniques and select the correct level of debug options when compiling your program.

  – Using service programs as a means of reusing code reduces the overall storage requirements for your application.

  – Generating DBGVIEW data may increase program object size significantly depending on the DBGVIEW options used.

    - If DBGVIEW(*LIST) compile option is chosen, the compile listing used for debugging is stored with the object, thus greatly increasing the program object. Carefully weigh the advantage of having a compiler listing stored with your object against the additional storage requirements.

- Consider using DGBVIEW(*SOURCE). It may give you similar capabilities in debug, but results in a smaller program object size.

13.4 Working Memory Guidelines for Compiles

Working memory size is the amount of memory required to do a task satisfactorily. Think of working memory size this way: Given infinite memory, the compiler will run at its optimal speed. If you restrict memory, the compiler will have to swap pages to DASD making it run slower. The more memory is restricted, the more time the compiler spends swapping memory pages. Working set size curves and elapsed time data for a small to medium sized OPM RPG program are shown in Figure 36 on page 232 and Table 76 on page 232. This program has 1500 C specifications and 5300 MI instructions. This program would compile reasonably quickly using an 8 MB pool if the program was not optimized. The optimized program′s compilation will benefit from as much memory as you can give it, although there is not much benefit beyond 64 MB.

There has been a significant reduction in working set size for optimized compiles on V3R7 when compared to V3R6. This reduction has two effects: a lower elapsed time for individual compiles due to fewer I/O operations, and a higher CPU utilization for these compiles (cpu time divided by elapsed time). A side effect of V3R7 is there is a smaller gain in throughput by running several compiles simultaneously. This technique could yield significant throughput improvements in V3R6 due to the lower CPU utilization of individual V3R6 compiles. The better utilization of V3R7 reduces the amount of idle CPU time available for simultaneous jobs.

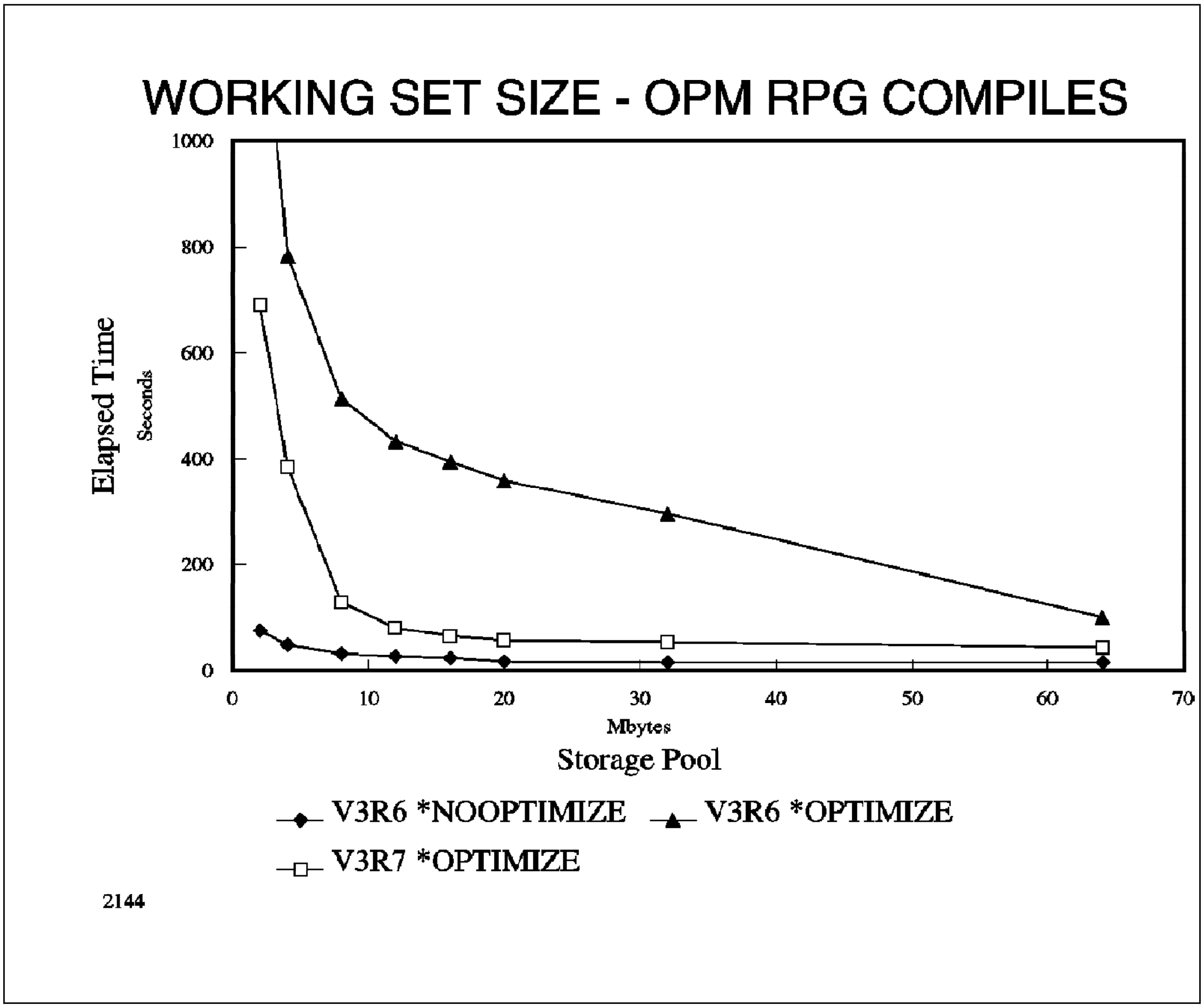


Figure 36. Working Set Size of OPM RPG Compiles

Table 76. Working Set Size of OPM RPG Compiles.

Pool Size (Mb)	Elapsed Time in Seconds		
	V3R6 *NOOPTIMIZE	V3R6 *OPTIMIZE	V3R7 *OPTIMIZE
2	76	1353	690
4	50	783	384
8	33	513	131
12	28	432	81
16	24	394	66
20	17	359	58
32	16	296	55
64	16	102	44
96	16	59	44

Note: These times were taken on a Model 510-2144

Conclusions/Recommendations

- As a general rule, to achieve minimum OPM and ILE compile times, use 16-20 MB to compile a medium-size program and 32MB to compile a large program. Smaller pool sizes will result in longer compile times, and less than 8MB is not recommended.
- Regarding minimum system configurations, a system with 32 MB of memory may be sufficient for casual application development work (for example infrequent compilations). This of course also depends on what other workload is running on the system. For systems that are used primarily for application development work, a minimum of 64MB of memory should be considered.
- As a general guideline for the memory size of systems used primarily for application development work, you should assume each concurrent compile requires about 25MB of main storage. For example, if the system needs to support 10 concurrent compiles, then as an initial estimate, the memory size of the system should be 256MB. If there is other work in addition to the application development work, the main storage requirements for that work needs to be taken into account also. For detail system capacity sizing, you should use BEST/1 for OS/400. BEST/1 takes into the account the additional main storage required for application development workloads, and should be used to accurately size main storage needs.

13.5 Application Compute Intensive Performance

In general, the performance improvement of applications referred to as ′application compute intensive′ is significantly more than the improvement of traditional commercial applications when moving from CISC to RISC technology.

In addition, V3R7 provides improvements to code generation over V3R6, especially for ILE C. The improvements can provide up to a 5 to 10% reduction in CPU time requirements for application programs and these improvements generally apply across the processor line. The greatest benefits come from recompiling programs rather than just retranslation.

This will cover:

- Performance of traditional commercial applications
- What is meant by ′application compute intensive′
- Why the performance of these application types improve significantly on V3R6

Traditional Commercial Applications

If you look at the CPU time profile for a traditional commercial application, typically up to 10-20% of the CPU time is spent in application programs, while the remaining 80+% is spent in operating system programs (Figure 37 on page 235). This is because traditional commercial applications spend much of their CPU time in system services such as database I/O, query processing, workstation/printer processing, and communication I/O.

Since a large part of the CPU time for commercial applications is spent in system services, performance improvements in these applications would result from either changing the application to utilize system services more efficiently, or from performance improvements to the system services. The performance of AS/400 system services has been optimized over several releases of OS/400 and perform equivalently on V3R6 as compared to V3R1. As a result, the performance of traditional commercial applications on V3R6 will be equivalent to V3R1 on a system with the same relative performance rating. For these types of applications, V3R6 offers improved price/performance and significant potential for performance growth over V3R1 with the introduction of PowerPC technology.

Application Compute Intensive

As compared to the traditional commercial applications, there are AS/400 applications where much more of the CPU time is spent in application programs (Figure 37 on page 235). These types of programs are referred to as ***Application Compute Intensive***. For example, applications that implement complex business rules for decision making are typically application compute intensive, as are financial modelling applications that do a significant number of numeric calculations.

Another example of applications that may be application compute intensive are the growing number of portable applications available on the AS/400. To achieve a high level of portability, these applications typically use only functions widely available on a number of systems. They implement their own functions which are provided more efficiently by unique OS/400 system services. As a result more CPU time is spent in application code.

The amount of performance improvement of applications that are application compute intensive depends on the actual workload, but you can expect to see improvements ranging up to 3 times when moving to V3R6.

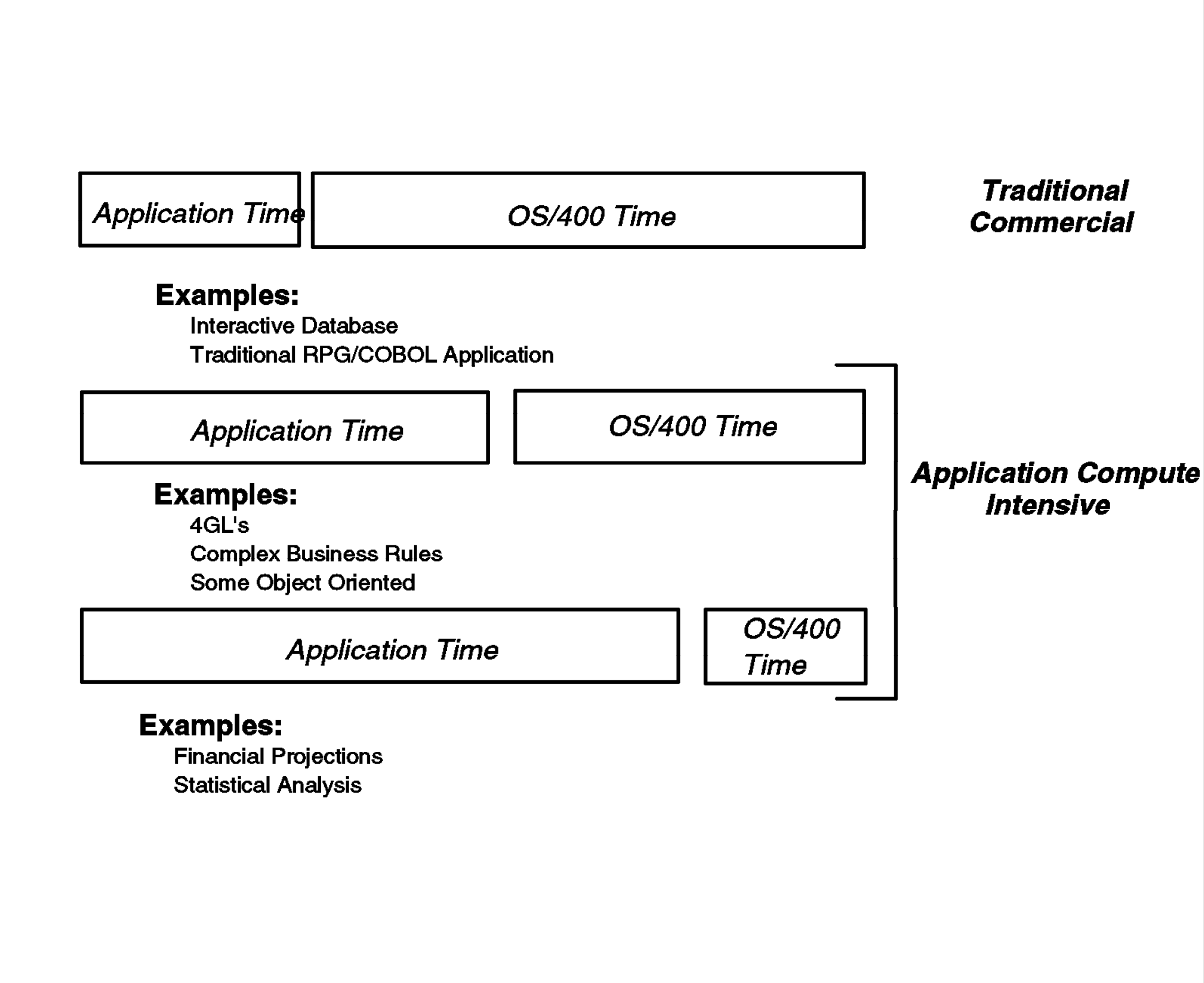


Figure 37. CPU Time Profile - Commercial versus Application Compute Intensive

ILE Compiler Optimization

Although applications that are application compute intensive can be written in other languages, they are typically written in ILE C. The main reasons for the significant performance improvements of these kinds of applications on V3R6 are:

- The nature of compute intensive applications in C are fundamentally different than that of traditional commercial applications. For example, they typically have more loop iterations, use pointers more intensively, and use integer as compared to decimal data types. These types of operations fit the RISC computation model more closely than code generated in commercial applications.
- ILE C coding paradigms give ILE more opportunities for optimization, such as strength reduction of loops, and common subexpression elimination.
- ILE can better exploit the power of RISC hardware. For example, the superscalar design of RISC provides multiple instruction pipelines which permit multiple instructions to be executed at the same time. The optimizing translator takes advantage of the superscalar design using instruction

scheduling to resequences instructions to maximize instruction overlap. This was not (generally) available on CISC.

Recommendations

- For general application programming tips and techniques refer to the *V3R1 Performance Management Redbook (GG24-3723-02).*
- ILE C applications that are application compute intensive should be compiled at optimization level 40 before putting them into production. This provides the highest level of optimization.
- New with ILE C/400 in V3R6 is direct mapping of stream I/O interfaces to the Integrated File System (POSIX) APIs. Optimum stream I/O performance is achieved when data is stored in the QOpenSys file system.

13.6 JAVA Performance

For information on JAVA performance on the AS/400, refer to the Java Performance (Chapter 7) section of this document or refer to the following Frequently Asked Question (FAQ) on the internet at URL:

```
http://www.softmall.ibm.com/as400/java/qanda.html
```

Chapter 14. DASD Performance

The change from a 512 byte to a 4KB page size will in general not noticeably change the DASD response time characteristics of an application as long as sufficient memory is added when upgrading to RISC. This is particularly true for batch applications that typically have I/O sizes that exceed 4KB. For applications such as interactive applications doing mostly random I/O, the 4 KB page size may decrease the number of I/O operations, depending on if additional data is accessed in the same 4KB page.

NOTE: The 4KB page size plus code expansion due to the RISC architecture will result in increased main storage requirements over IMPI. Refer to 18.3, "Main Storage Sizing Guidelines" on page 297 for a discussion on how much main storage is required on RISC as compared to IMPI.

14.1 Device Performance Characteristics

This section compares the performance of the Internal DASD Subsystems based on the 65x2 RAID Controllers or 6530 Storage Controller with the external 9337 Disk Array Subsystem using a system configured with an equivalent amount of DASD capacity. This section also contains performance characteristics for the 6532, new 6533, 2726 and new 2740 and 2741 RAID Controllers and 6751, new 6754 and 9728 MFIOPs. The performance is based on measurements and modeling done in the development laboratory. Because the performance of the AS/400 system is dependent on many factors, these characteristics are very general in nature. To assess the various configuration options, one of the capacity planning tools should be used.

The performance characteristics of Internal DASD is listed in Table 77 on page 239 and Table 78 on page 240 and the performance characteristics of External DASD is listed in Table 79 on page 241. One new DASD model (6714 - 18GB capacity) has been announced for V4R2. The tables do not list all of the feature codes, but it does provide performance information for most of the disk configurations. For example, the 6522 IOP has the same performance characteristics as the 6502 IOP. For a description of the DASD models supported by the 6502, 6512, 6530, 6532, 6533, 6751, 6754, 2726, 2740, 2741 and 9728 IOP/IOAs refer to Appendix C, "DASD IOP Device Characteristics" on page 321.

In the tables, the following measures of performance are listed.

Service Time is the time required to perform the ″Interactive op″ described in the next paragraph. The time starts with the request from the CPU to the Disk IOP and the time stops when the data is in main storage (read) or when the data is on the disk or in the write cache (write). Queueing time is not included.

Interactive Ops/Sec is an estimate of the number of IOs that can be done at 40% utilization using the service time calculated for the previous column. If the disk model contains 2 arms, this number only reflects the capacity of one arm. At 40%-50% utilization, the disk arms are at the ″knee of the curve″. As utilization exceeds the ″knee of the curve″, response time increases significantly and becomes erratic. We assume the following:

- 40% arm utilization
- 7KB transfer size

- 70% read and 30% write
- 80% 1/3 seek and 20% 0 seek

Interactive Rel is the Relative Interactive performance of the disk drives. This column is the same as the INTERACTIVE Ops/Sec column except that the numbers are normalized to 1.0.

Batch Hours is an estimate of how long batch type applications would execute. The duration of many batch type jobs depends on the performance of the disk. For ease of understanding, the numbers are normalized to 8 hours assuming the slowest disk drive is used. We assume the following:

- 75% of the batch job time is disk IO
- Average of 4KB, 8KB and 16KB transfer sizes
- 70% read and 30% write
- 20% 1/3 seek and 80% 0 seek

Ops/Sec/GB is an estimate of how many system physical disk IOs per second per usable GB of space that the specific model of DASD can perform when the arm is 40% utilized. The write cache effectiveness reduces the volume of writes that the physical disk drive must support. For the 9337-2xx models, the write cache effectiveness is assumed to be 45% and for the 9337-4xx and 9337-5xx models it is assumed to be 65%. For the 6502 IOP, the write cache effectiveness is assumed to be 55%. For the 6512, 6532, 6533, 6751, 6754, 2726, 2740 and 2741 IOP/IOAs, the write cache effectiveness is assumed to be 65%. We use the service time required to physically write the record to DASD. The service time contained in column four included the faster write completions that resulted when the write was safely in the write cache.

Table 77. DASD Performance - Internal DASD

Disk Model	MB	Number of Arms	Service Time	Interactive		Batch Hours	Ops/Sec/GB	
				Ops/Sec	Rel		Base	RAID
6502-6605	1031	1	8.7	46.0	2.3	3.6*	38	29
6502-6606	1967	1	8.8	45.5	2.3	3.6*	20	15
6502-6607	4194	1	8.8	45.5	2.3	3.6*	9	7
6502-6713	8589	1	9.1	44.0	2.2	3.6*	4	3
6502-6714	17548	1	9.1	44.0	2.2	3.6*	2	2
6512-6605	1031	1	8.2	48.8	2.4	3.4*	43	36
6512-6606	1967	1	8.3	48.2	2.4	3.4*	22	18
6512-6607	4194	1	8.3	48.2	2.4	3.4*	10	8
6512-6713	8589	1	8.6	46.5	2.3	3.4*	5	4
6512-6714	17548	1	8.6	46.5	2.3	3.4*	2	2
6530-6605	1031	1	11.4	35.1	1.7	3.9	34	
6530-6606	1967	1	11.6	34.5	1.7	3.9	18	
6530-6607	4194	1	11.6	34.5	1.7	3.9	8	
6530-6713	8589	1	12.0	33.3	1.7	4.0	4	
6530-6714	17548	1	12.0	33.3	1.7	4.0	2	

Note:

The 6502 and 6512 IOP write cache is only used for 1 GB and larger DASD. The write cache is NOT used for any 400MB or 988MB DASD that are attached.

* For the 6502 and 6512 IOPs in RAID mode, most batch jobs will run nearly as fast as if they were run in ′base′ or mirrored mode. Only in extreme cases will the RAID mode cause degradation. An example is when there are sequences of hundreds of writes to a single IOP in a short period of time.

Table 78. DASD Performance - Internal DASD (continued)

Disk Model	MB	Number of Arms	Service Time	Interactive		Batch Hours	Ops/Sec/GB	
				Ops/Sec	Rel		Base	RAID
6533-6605	1031	1	8.0	50.0	2.5	3.4*	43	36
6533-6606	1967	1	8.1	49.4	2.5	3.4*	23	19
6533-6607	4194	1	8.1	49.4	2.5	3.4*	11	9
6533-6713	8589	1	8.4	47.6	2.4	3.4*	5	4
6533-6714	17548	1	8.4	47.6	2.4	3.4*	2	2
2741-6605	1031	1	8.0	50.0	2.5	3.4*	43	36
2741-6606	1967	1	8.1	49.4	2.5	3.4*	23	19
2741-6607	4194	1	8.1	49.4	2.5	3.4*	11	9
2741-6713	8589	1	8.4	47.6	2.4	3.4*	5	4
2741-6714	17548	1	8.4	47.6	2.4	3.4*	2	2
6754-6605	1031	1	8.0	50.0	2.5	3.4*	43	36
6754-6606	1967	1	8.1	49.4	2.5	3.4*	23	19
6754-6607	4194	1	8.1	49.4	2.5	3.4*	11	9
6754-6713	8589	1	8.4	47.6	2.4	3.4*	5	4
6754-6714	17548	1	8.4	47.6	2.4	3.4*	2	2
9728-6605	1031	1	10.9	36.7	1.8	3.8	38	
9728-6606	1967	1	11.0	36.4	1.8	3.8	18	
9728-6607	4194	1	11.0	36.4	1.8	3.8	9	
9728-6713	8589	1	11.3	35.4	1.8	3.9	4	
9728-6714	17548	1	11.3	35.4	1.8	3.9	2	

Note:

* The new 6533 IOP has slightly better performance than the 6532 IOP but will usually be noticeable only at higher throughput ranges. The new 2741 IOA has slightly better performance than the 2726 IOA but will usually be noticeable only at higher throughput ranges. The new 6754 MFIOP has the same performance relationship with the 6751 MFIOP. The ***new 2740 IOA*** (which is targeted for smaller systems) has similar performance to the 2726 IOA over typical operating ranges, but has slightly slower performance at higher throughput ranges.

* For the 6532, 6533, 6751, 6754, 2726, 2740 and 2741 IOP/IOAs in RAID mode, most batch jobs will run nearly as fast as if they were run in ′base′ or mirrored mode. Only in extreme cases will the RAID mode cause degradation. An example is when there are sequences of hundreds of writes to a single IOP in a short period of time.

These IOP/IOAs are also capable of attaching Ultra-SCSI (40 MB/sec bus) versions of the 6606, 6607, 6713 and 6714 DASDs. These devices can improve performance for workloads characterized by large disk I/O operations.

Table 79. DASD Performance - External DASD

Disk Model	MB	Number of Arms	Service Time	Interactive		Batch Hours	Ops/Sec/GB	
				Ops/Sec	Rel		Base	HA
9337-210	1084	2	12.4	32.3	1.6	4.3*	50	36
9337-215	1084	2	9.8	40.8	2.0	3.9*	63	46
9337-220	1940	2	12.5	32.0	1.6	4.3*	28	20
9337-225	1940	2	11.0	36.4	1.8	4.0*	32	23
9337-240	7868	4	11.3	35.4	1.8	4.0*	15	11
9337-420	3880	4	8.6	46.5	2.3	3.5*	43	36
9337-440	7868	4	8.8	45.5	2.3	3.5*	21	18
9337-480	16776	4	9.1	44.0	2.2	3.6*	10	8
9337-540	7868	4	8.6	46.5	2.3	3.5*	21	18
9337-580	16776	4	8.6	46.5	2.3	3.5*	10	8
9337-590	34356	4	8.9	44.9	2.2	3.6*	5	4

Note:

* For the 9337-2xx, 9337-4xx and 9337-5xx models in HA mode, most batch jobs will run nearly as fast as if they were run in ′base′ or mirrored mode. Only in extreme cases will the RAID mode cause degradation. An example is when there are sequences of hundreds of writes to a single IOP in a short period of time.

Conclusions / Recommendations

The 6532, 6751 and 2726 DASD IOP/IOAs have similar performance characteristics. The new 6533 IOP, 2741 IOA and 6754 MFIOP have slightly better performance characteristics, which are more beneficial at higher throughput ranges. The new 2740 IOA (which is targeted for smaller systems) has similar performance characteristics to the 2726 IOA over typical operating ranges, but has slightly slower performance at higher throughput ranges. These IOP/IOAs are also capable of attaching Ultra-SCSI (40 MB/sec bus) versions of the 6606, 6607 and 6713 DASDs. These devices can improve performance for workloads characterized by large disk I/O operations.

The DASD that are used in the Internal DASD Subsystems have read ahead buffers that can provide performance advantages. Like the 9337, each of these DASD has a 512K buffer. The buffer is allocated into multiple segments that are larger than 32K each. Read ahead data from recent IOs are kept in these buffer segments. Depending on the data access patterns, it is possible that the data needed is already contained in a buffer segment. If so, no physical access to the DASD is required. Depending on your data access patterns, this can significantly improve performance. Our analysis of several specific customer installations indicates that 10% to 30% of their DASD IO for ″interactive″ transactions would have already been contained in the read ahead buffer. For ″batch″ type jobs, 25% to 45% of their DASD IO would have already been contained in the read ahead buffer. The RAMP-C workload being used in this section has less than 10% of it′s DASD IOs already in the read ahead buffer.

For the 9337-2xx, 9337-4xx, 9337-5xx, 65x2 (also 2726, 2740, 2741, 6533, 6751 and 6754) models in HA mode, most batch jobs will run nearly as fast as if they were run in ″base″ mode or mirrored mode. Only in extreme cases will the HA mode cause degradation. An example of the extreme case is when there are sequences of hundreds of writes to a single 9337 or 65x2 in a short period of time.

You must ensure that you have enough arms to support the volume of DASD IOs that your customer will require. In some situations, using the larger capacity DASD may result in an insufficient number of arms to handle the required DASD IO volume. The Capacity Planning tools should be used to verify your configuration.

The recommended threshold for maximum DASD utilization for 1 arm configurations is higher than the threshold for multiple arm configurations. The reason for the lower recommendation for multiple arms is that it is assumed that when 2 or more arms have an average utilization of 40%, some of the arms may be at the 50% - 55% range while others will be lower. QSIZE400 and BEST/1 allow a 1 arm configuration to reach 55% before they recommend that an additional DASD be added.

Consider the following example. Assume you are configuring a system and need approximately 4000 MB of DASD space. You have the choice of 4 x 988MB or 2 x 1967MB. The 4 x 988MB configuration will support approximately 70% more DASD IOs as the 2 x 1967MB configuration. Because there is a maximum number of DASD devices that can be attached to each model, using the larger drives will allow more MB of DASD to be configured on your system.

The Performance Monitor (STRPFRMON command) captures additional performance data (buffer hits, etc.) for the 65x2, 6533 and 6530 attached DASD. This data is available in the QAPMDISK performance data file and is documented in Appendix A of the *AS/400 Work Management V4R3* (SC41-5306-02).

14.2 DASD Performance - Interactive

The implementation of the 4KB page size on RISC will improve system DASD IO efficiency. As a result of the larger page size, some DASD subsystem interactive Ops/Sec/GB ranges will appear lower than on IMPI.

Some DASD system performance charts included for RISC may differ from similar charts published for IMPI. These performance differences can be attributed primarily to the following:

- Differences in system processor power
- Differences in main storage configurations
- Differences in system page size
- Differences in allocation of data and programs on DASD.

Therefore, direct comparisons between RISC and IMPI system DASD performance charts are not recommended.

DASD Subsystem Performance - Base or Mirrored

The following bar graphs compare the service times for the AS/400 DASD subsystem offerings. The IO operations being performed are 7KB transfer size, 70% are reads and 30% are writes, and 80% require a seek over 1/3 of the disk surface while 20% require no seek. Queueing time is not included.

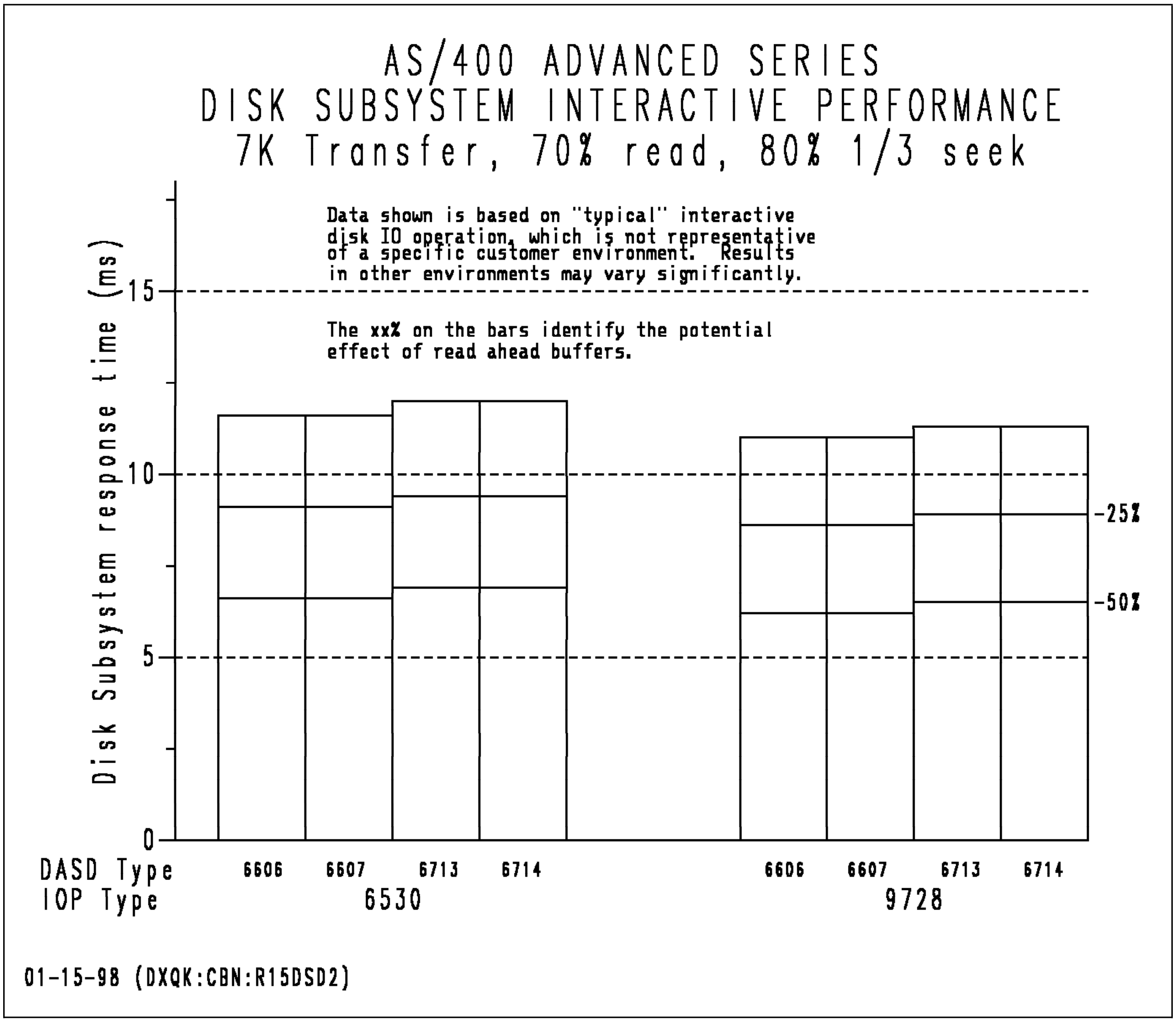


Figure 38. DASD Subsystem Performance / Non-RAID Capable - Base Mode

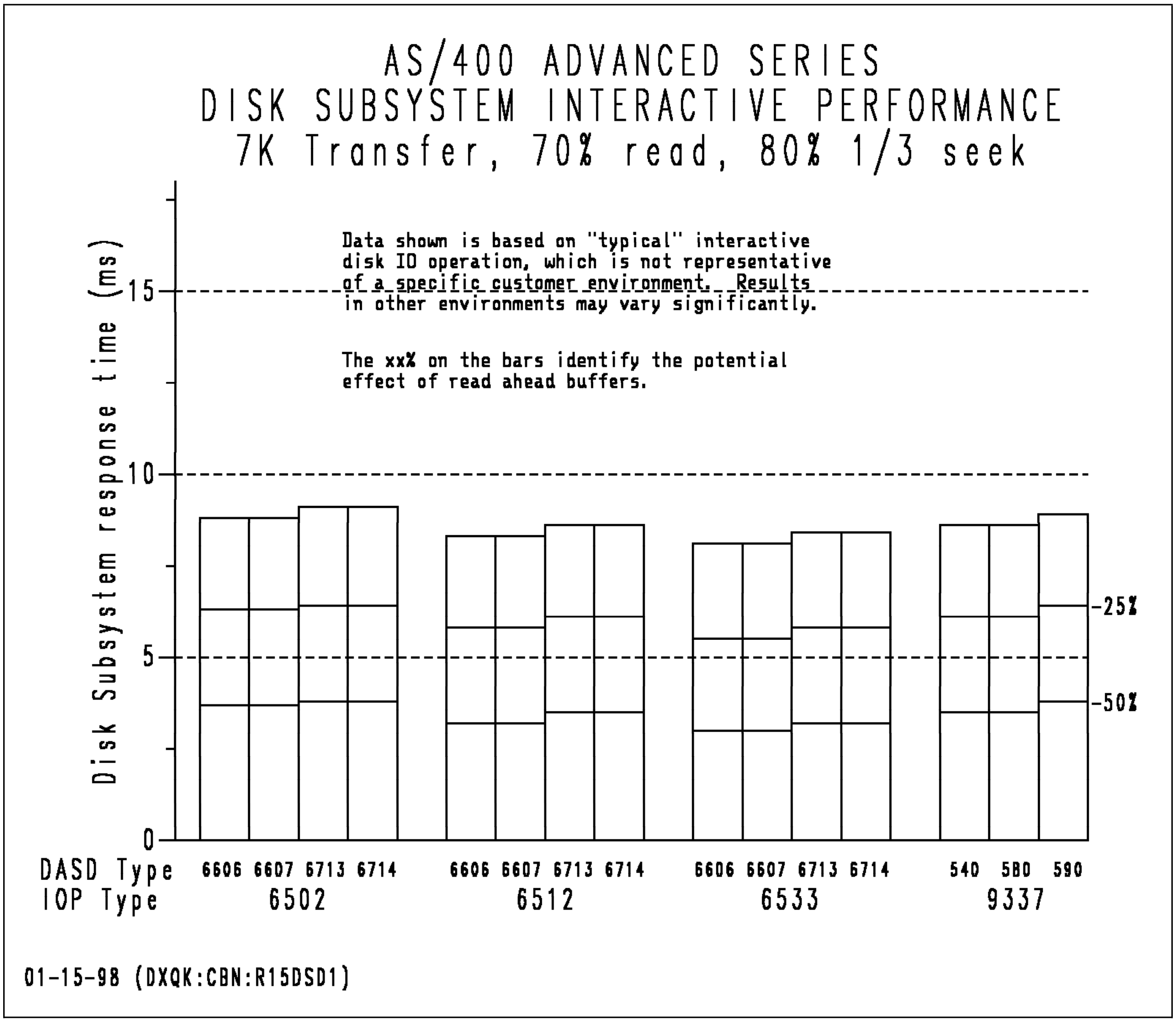


Figure 39. DASD Subsystem Performance / RAID Capable - Base Mode

Conclusions / Recommendations

- The performance of 6606, 6607, 6713 and 6714 disks is faster with the 6533 IOP than with the 6512 IOP. The 6754, 6751, 6532, 2726, 2740 and 2741 DASD IOP/IOAs have performance characteristics similar to the 6533 IOP over typical operating ranges.
- The performance with 6606, 6607, 6713 and 6714 disks is faster than the previous DASD types for all the subsystems.
- The 9728 subsystem performance is faster than the 6530 subsystem for the same type of DASD.
- The 6512 subsystem performance is better than the 6502 subsystem for the same type of DASD. This is due primarily to a faster processor and a larger 4 MB Write cache in the 6512.
- The 6512 subsystem performance is slightly better than the 9337-5xx subsystem for the same type of DASD.

- The 6502 subsystem performance is significantly better (32%) than the 6530 subsystem for the same type of DASD. This is due primarily to the 2 MB Write cache in the 6502.
- ″RAID Capable″ DASD subsystems are faster in base mode than ″Non-RAID Capable″ due to their write cache.
- The potential effect of read-ahead buffers are shown for the cases of having 25% and 50% of the total disk operations already in the read ahead buffer. Depending on the data access patterns, the buffers may provide significant performance improvements.
- The above conclusions hold for batch environments also. For actual batch performance results refer to Table 77 on page 239, Table 78 on page 240 and Table 79 on page 241.

AS/400 System Interactive Performance - Base

The following graph compares the relative interactive performance of an AS/400 model 510/2144 configured with 33.5GB of internal or external DASD. The internal load source drive was ignored for this comparison chart. The curves characterize what may occur on either a ′Base′ configuration or a mirrored configuration. The graph compares the 9337-580 model with the 65x2-4GB models for a commercial interactive environment.

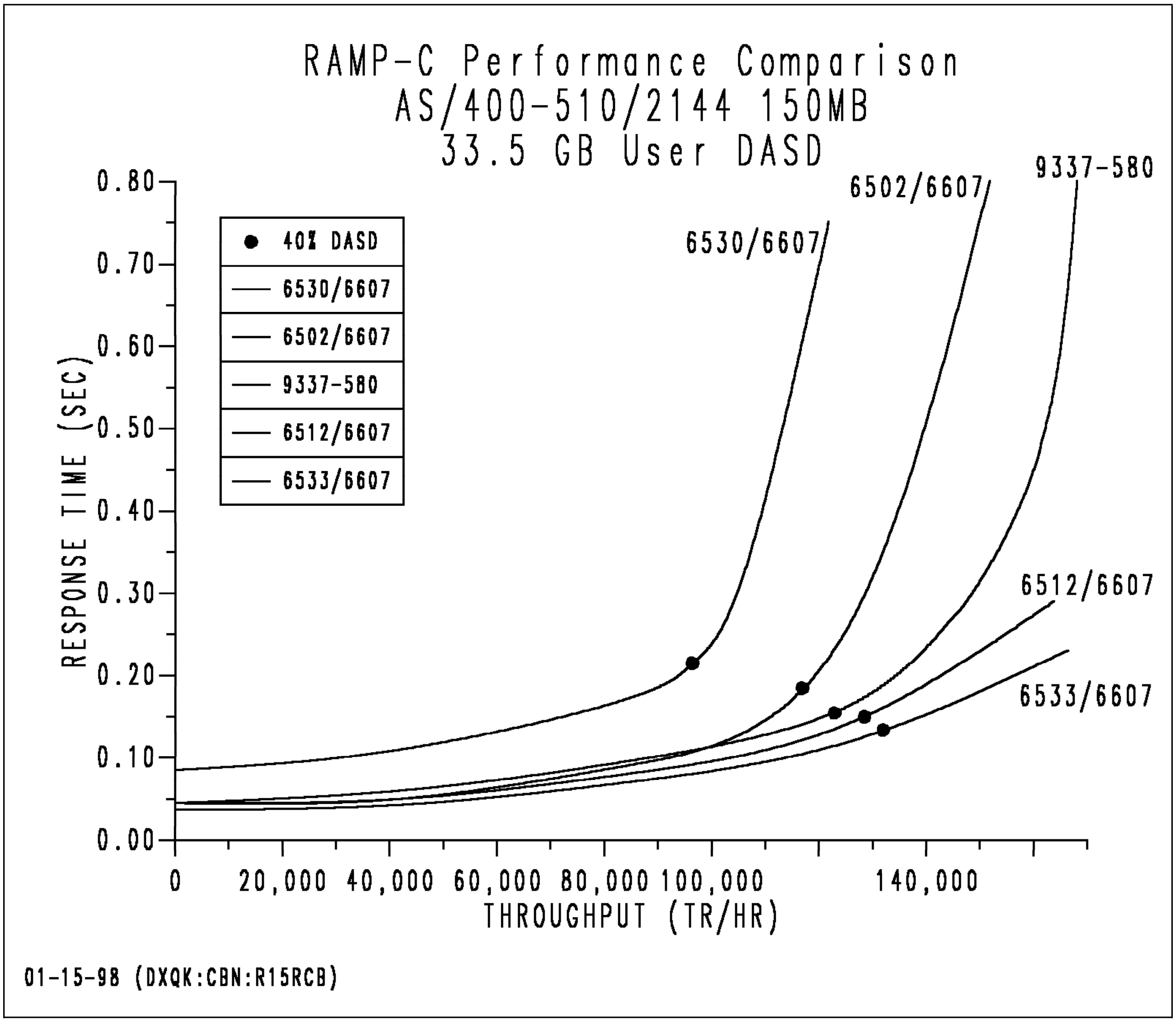


Figure 40. System Interactive Performance - Base Mode

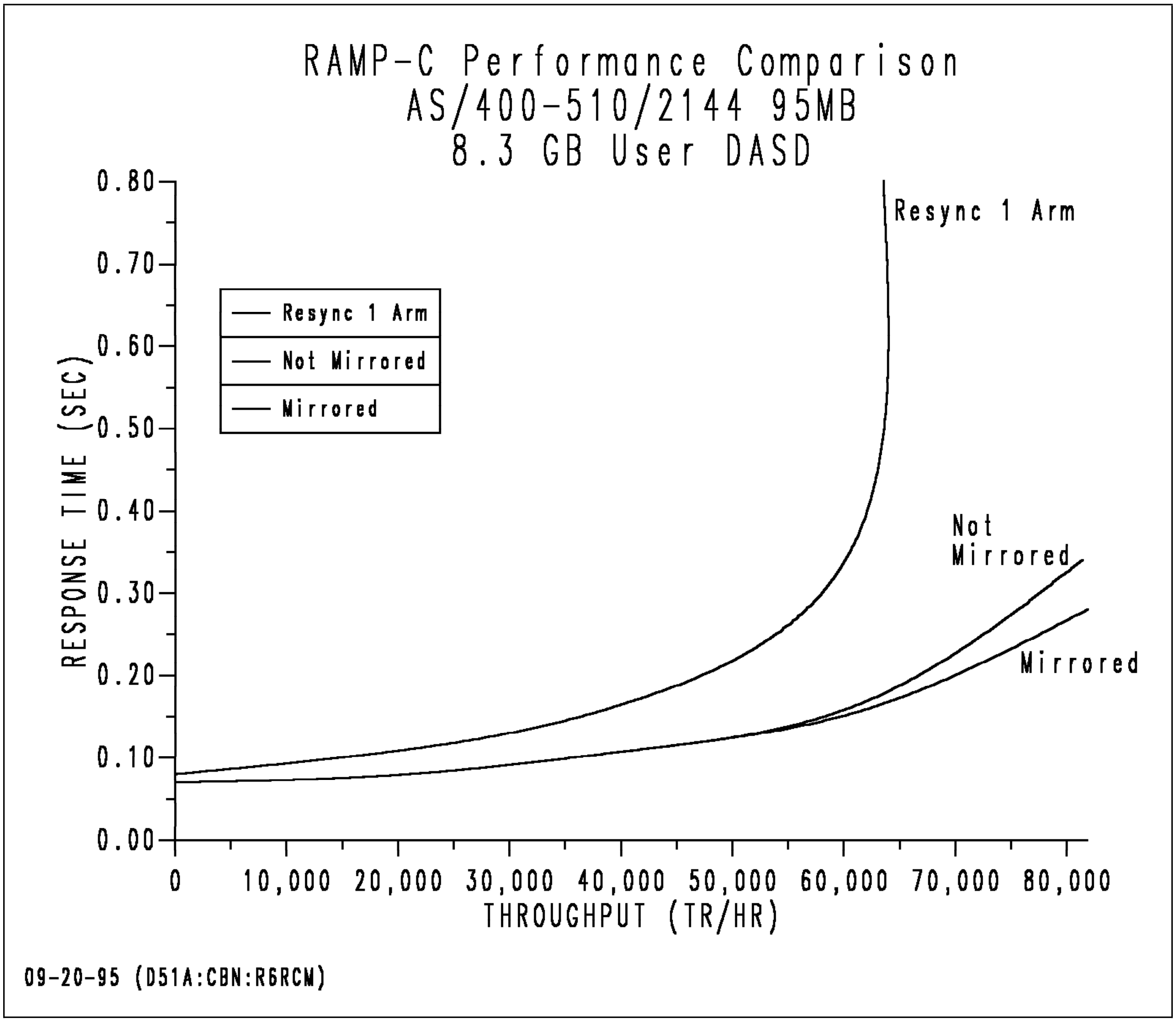
Conclusions / Recommendations

- The 6533/6607 DASD provides better interactive performance than the 6512/6607 DASD. The 6754, 6751, 6532, 2726, 2740 and 2741 DASD IOP/IOAs have performance characteristics similar to the 6533 IOP over typical operating ranges.
- The 6512/6607 DASD provides better interactive performance than the 9337-580 DASD.
- The 6512/6607 DASD provides better interactive performance than the 6502/6607 DASD, especially at higher system throughput.
- Performance with the 6502/6607 DASD is slightly better than the 9337-580 DASD for lower throughput, but becomes worse at higher throughputs.
- The 65x2 and 9337-5xx configurations have reduced volumes of physical disk IO due to the write cache. The write cache also greatly improves the service time for write ops.

- The 65x2 write cache provides a significant performance advantage over the 6530. When a write is requested to a 65x2, the 65x2 writes the data to the write cache and to the nonvolatile cache backup and the application is allowed to continue. Through a combination of the write cache and 65x2 nonvolatile memory, the 65x2 ensures the integrity of the data even if a failure should occur.
- This graph is based on RAMP-C workload. Other environments may vary significantly.
- The RAMP-C benchmark′s data access patterns are intentionally random, therefore, the read-ahead buffers provided only minimal benefit for RAMP-C. Depending on your data access patterns, the DASD read ahead buffers may provide significant performance improvements.
- Similar results may occur on other AS/400 models. Response time / throughput curves encounter a ″knee″ when a resource is used too heavily. CPU, main memory, IOP Processor and DASD are examples of resources that can cause ″knees″. If faster AS/400 CPUs are used, and other resources are unchanged, the possibility that memory or DASD will constrain the throughput increases. The BEST-1 Capacity Planner should be used to determine appropriate configurations.

AS/400 System Interactive Performance - Mirrored versus Base

The following graph compares the relative interactive performance of an AS/400 model 510/2144 configured with 8.4 GB of User DASD. The graph compares a mirrored environment with 16 arms to a base (not mirrored) environment with 8 arms. It also shows the system performance effects during the resync of a single arm.



09-20-95 (D51A:CBN:R6RCM)

Figure 41. System Interactive Performance - Mirrored versus Base - Internal DASD

Conclusions / Recommendations

- The mirrored configuration provides equal or better interactive performance than the base configuration (not mirrored). The better mirrored performance is due to having more arms to handle the larger number of read ops at higher throughput.
- The system performance is less during the time it takes to resync an arm, especially at higher throughput. The customer could choose to schedule the resync during a period of lower system activity or quiesce some applications

during the resync time (20 to 40 minutes for a 1GB device). Larger devices will have proportionally longer resync time.

- This graph is based on RAMP-C workload. Other environments may vary significantly.

AS/400 System Interactive Performance - RAID

The following graph compares the 65x2 RAID DASD Subsystems with the 9337-580 HA Subsystem. All subsystems contained 8 4GB arms.

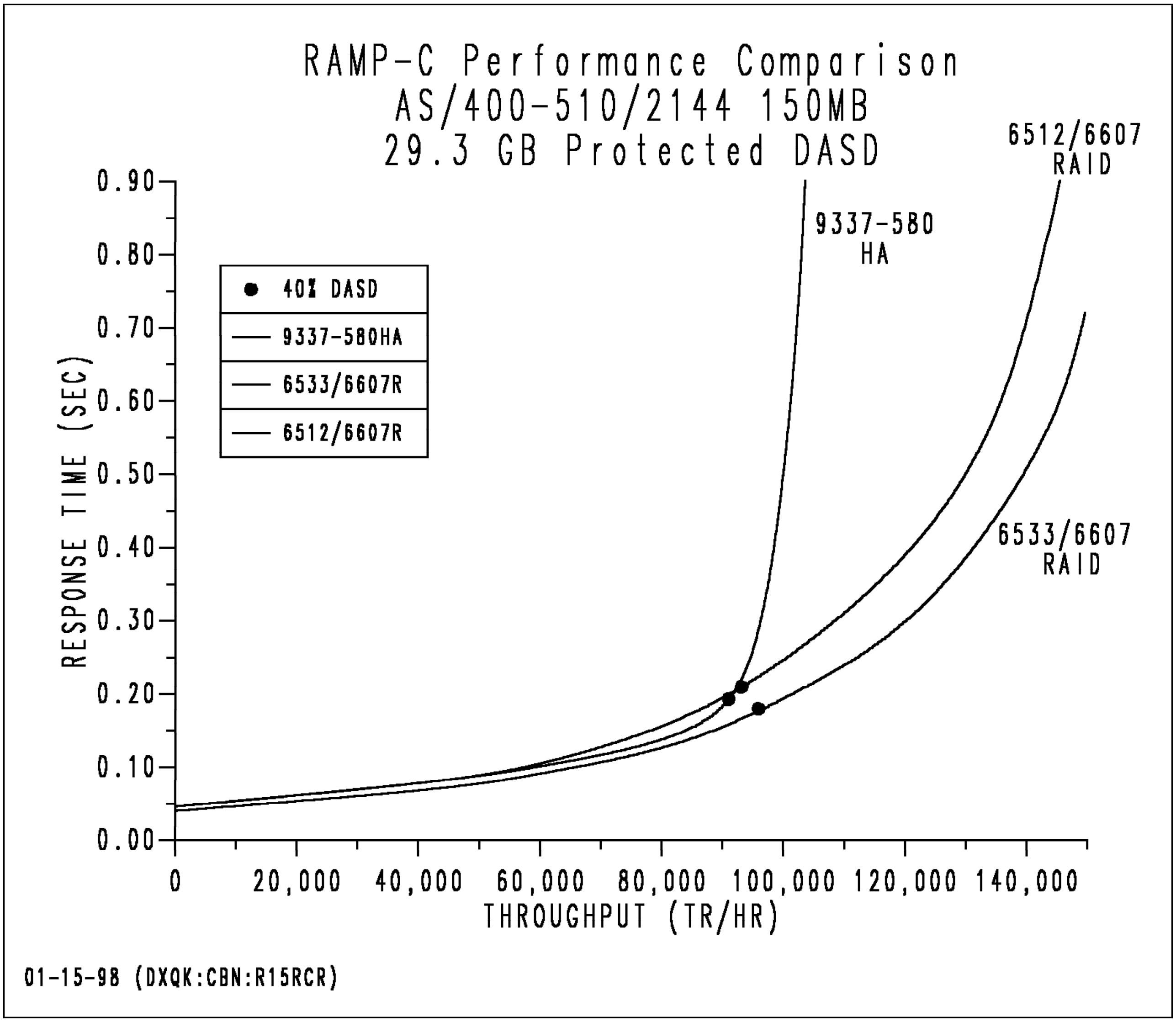


Figure 42. System Interactive Performance - RAID Mode

Conclusions / Recommendations

- The 6533/6607 RAID DASD provides better interactive performance than the 6512/6607 RAID DASD. The 6754, 6751, 6532, 2726, 2740 and 2741 DASD IOP/IOAs have performance characteristics similar to the 6533 IOP over typical operating ranges.
- Performance with the 9337-580 HA is comparable with the 6512/6607 RAID DASD for similar throughput.

- The 9337 measurements were done with 4 parity arms per array and the 65x2 measurements were done with 8 parity arms per array. In general 8 parity arms per array will provide better performance at higher throughputs. At low to medium throughput, there is little performance difference between 4 and 8 parity arms per an 8 arm array. On the 65x2 (also 2726, 2740, 2741, 6533, 6751 and 6754), parity arrays of 8 or more arms should be configured with 8 parity arms if possible.

Impact of failed DASD in RAID Subsystem

This is a general discussion of RAID-5.

- 65x2 (also 2726, 2740, 2741, 6533, 6751 and 6754) RAID and 9337 HA DASD subsystems let the AS/400 system continue to operate even after a single DASD failure.
- With system checksum, a DASD failure will cause the AS/400 system to stop and an IPL will be required.
- RAID-5 overhead can become significant when a DASD fails.

  READ

  - To read from a failed DASD, RAID-5 must read ALL remaining arms in the set. (This means anywhere from 3 to 9 overlapped reads, where 1 was sufficient before). This will have a significant effect on **the failed** DASD subsystem throughput and response time. This degraded mode will last until the DASD is repaired and the ″rebuild″ of the failed DASD′s parity stripes are complete.
  - Reads to other DASD on the same subsystem are unaffected.
  - Bottom line, if the parity array has 4 arms, this results in 1.5 times increase in DASD IO read volume to this array. If the array has 8 arms, the result is 1.75 times increase in DASD IO read volumes to this array.

  WRITE

  - There are 3 separate scenarios that apply to RAID-5 writes with one failed DASD.
    - If the failed DASD is not involved (either for data or for the checksum stripe), the writes are handled as normal RAID-5 writes. (2 reads plus 2 writes)
    - With a write to a failed DASD, all remaining DASD in the set must be read and then one write will be done to the checksum stripe. (N-1 reads plus 1 write, N=number of DASD arms)
    - If the DASD that contains the checksum stripe is the failed DASD, then all that is required is a write to the DASD that contains your data. (1 write)
    - Bottom line, if the parity array has 4 arms, each write averages a 3.25 increase in DASD IO write volume to this array. If the array has 8 arms, the result is a 4.13 times increase in DASD IO write volumes to this array.

  General discussion

  - When running in ″exposed″ mode, the fewer the number of arms in each parity array, the smaller the degradation.
  - On systems with smaller amounts of DASD capacity, the degradation will be more noticeable. This is because there are fewer arrays which means that a larger percentage of the DASD operations will be directed to the ″exposed″ array.
  - The DASD IO to any subsystems that do not have a failed DASD are unaffected.

- If the Customer cannot tolerate the temporary performance degradation that would occur with a RAID-5 DASD failure, they should consider mirroring.
- To obtain acceptable performance with a failed RAID-5 DASD, some customers may have to delay nonessential work until after the DASD is repaired. For example, a customer may continue to process their on-line order entries but delay their office tasks.
- In configurations with small amounts of total DASD space and with high availability requirements, mirroring may be a more satisfactory option
- The estimated time to rebuild a DASD is approximately 30 minutes for a 8 arm array on a dedicated system with no other jobs running. If other concurrent jobs being run on the system are requesting 130 IOs per second to this DASD subsystem, the rebuild time will increase to approximately 1 hour.

The following chart compares the impact of one failed DASD on several configurations.

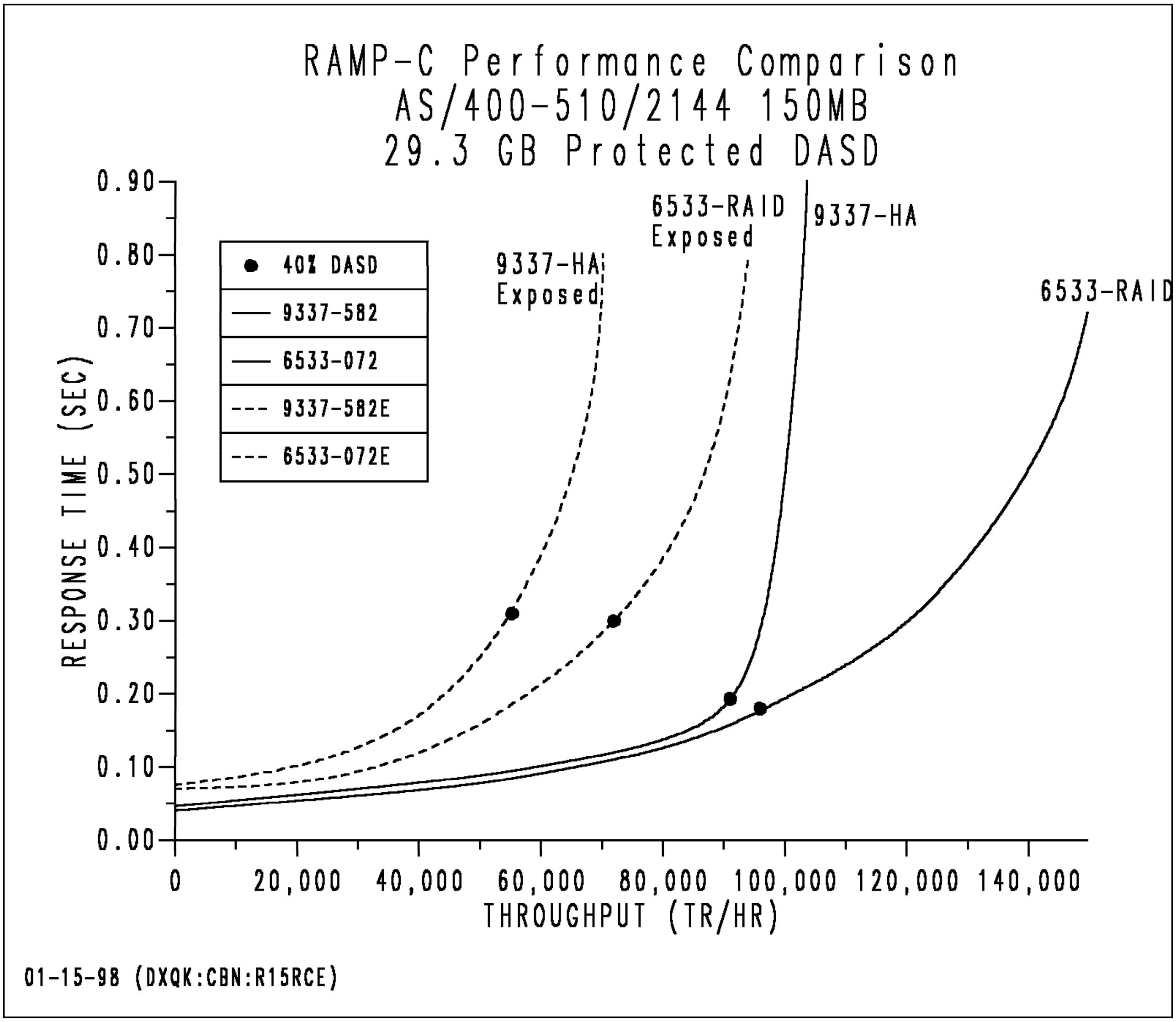


Figure 43. Performance Impact of Failed DASD

Conclusions / Recommendations

- The 6533 performs better in ″exposed″ mode than the 9337-580 due to the ability of the 6533 to handle higher throughput more efficiently. The 6754, 6751, 6532, 2726, 2740 and 2741 DASD IOP/IOAs have performance characteristics similar to the 6533 IOP over typical operating ranges.
- With a parity array of 8 arms on a 6533, ″exposed″ mode throughput is about half of normal throughput.
- If there were ″n″ 65x2s in the configuration, 1/n of the DASD IOs are to the ″exposed″ 65x2.
- As ″n″ gets larger, the impact of a DASD failure to overall system performance is reduced.
- Additional degradations occur during rebuild for both 65x2 and 9337. The rebuild can be scheduled for periods of lower system utilization.

Ops/Sec/GB Guidelines for DASD Subsystems

The metric used in determining DASD subsystem performance requirements is the number of I/O operations per second per installed GB of DASD (Ops/Sec/GB). Ops/Sec/GB is a measurement of throughput per actuator. Since DASD devices have different capacities per actuator, Ops/Sec/GB is used to normalize throughput for different capacities. An Ops/Sec/GB range has been established for each DASD type so that if the DASD subsystem performance is within the established range, the average arm percent busy will meet the guideline of not exceeding 40%.

The implementation of the 4KB page size on RISC will improve system DASD IO efficiency. As a result of the larger page size, some DASD subsystem interactive Ops/Sec/GB ranges will appear lower than IMPI.

The following bar charts show the ″rule of thumb″ for the Physical system Ops/Sec/GB of usable space that internal DASD subsystems can achieve with various DASD types. (To compute usable GB, we assume that the DASD subsystems have 8 disk units installed). The top of each bar is the volume of 7K transfer, 80% 1/3 seek, 30% write operations that each model can achieve when it is 40% busy. For the 6502, we assume that the write cache has an efficiency of 55%. For the 6512, 6532, 6533, 6751, 6754, 2726, 2740 and 2741 we assume that the write cache has an efficiency of 65%. The vertical scale is the volume of physical Ops issued from the system. The 65x2 RAID disk subsystem bars are lower because of the additional work that these subsystems must do to maintain the RAID-5 parity stripes.

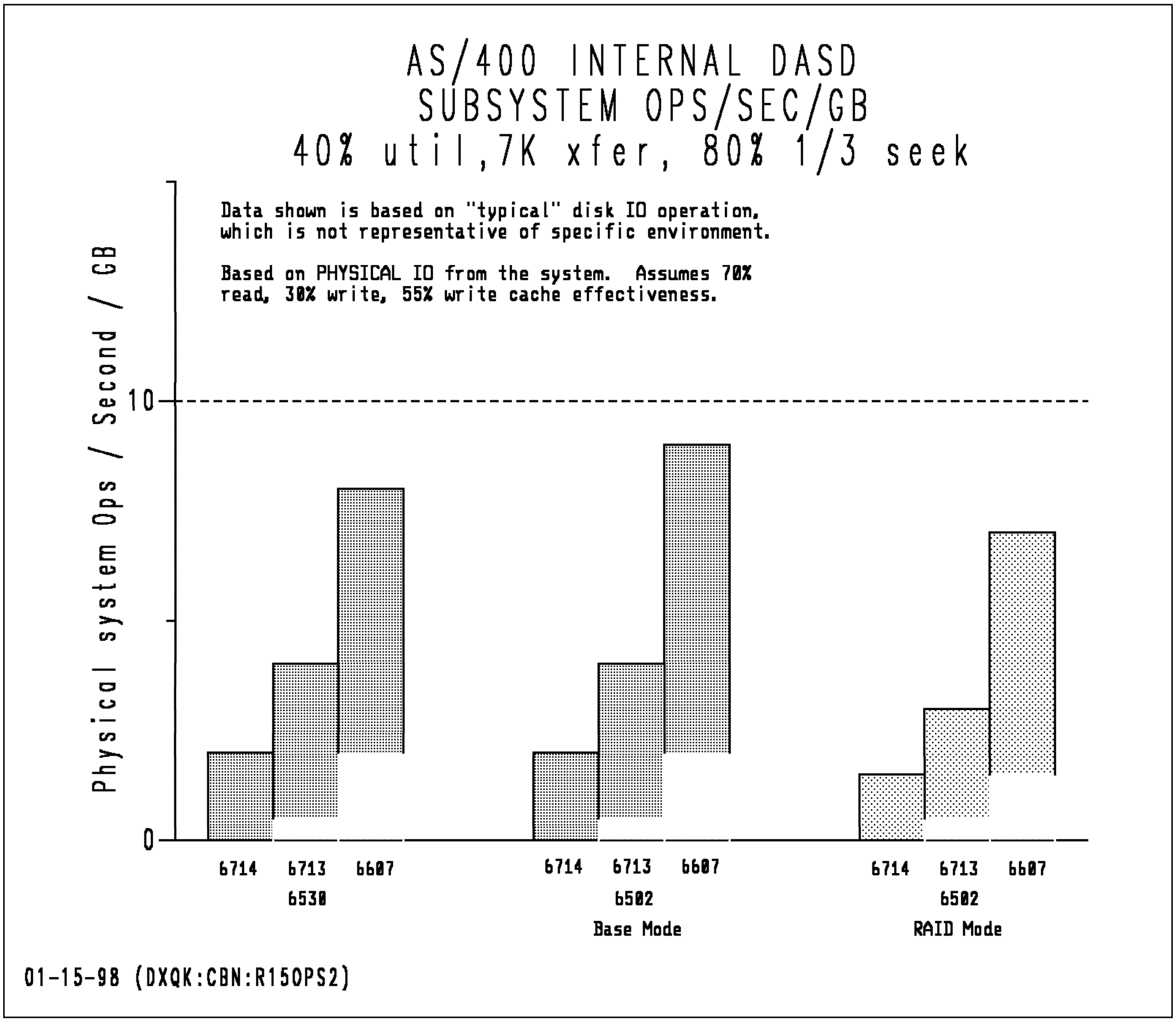


Figure 44. Ops/Sec/GB - Internal DASD

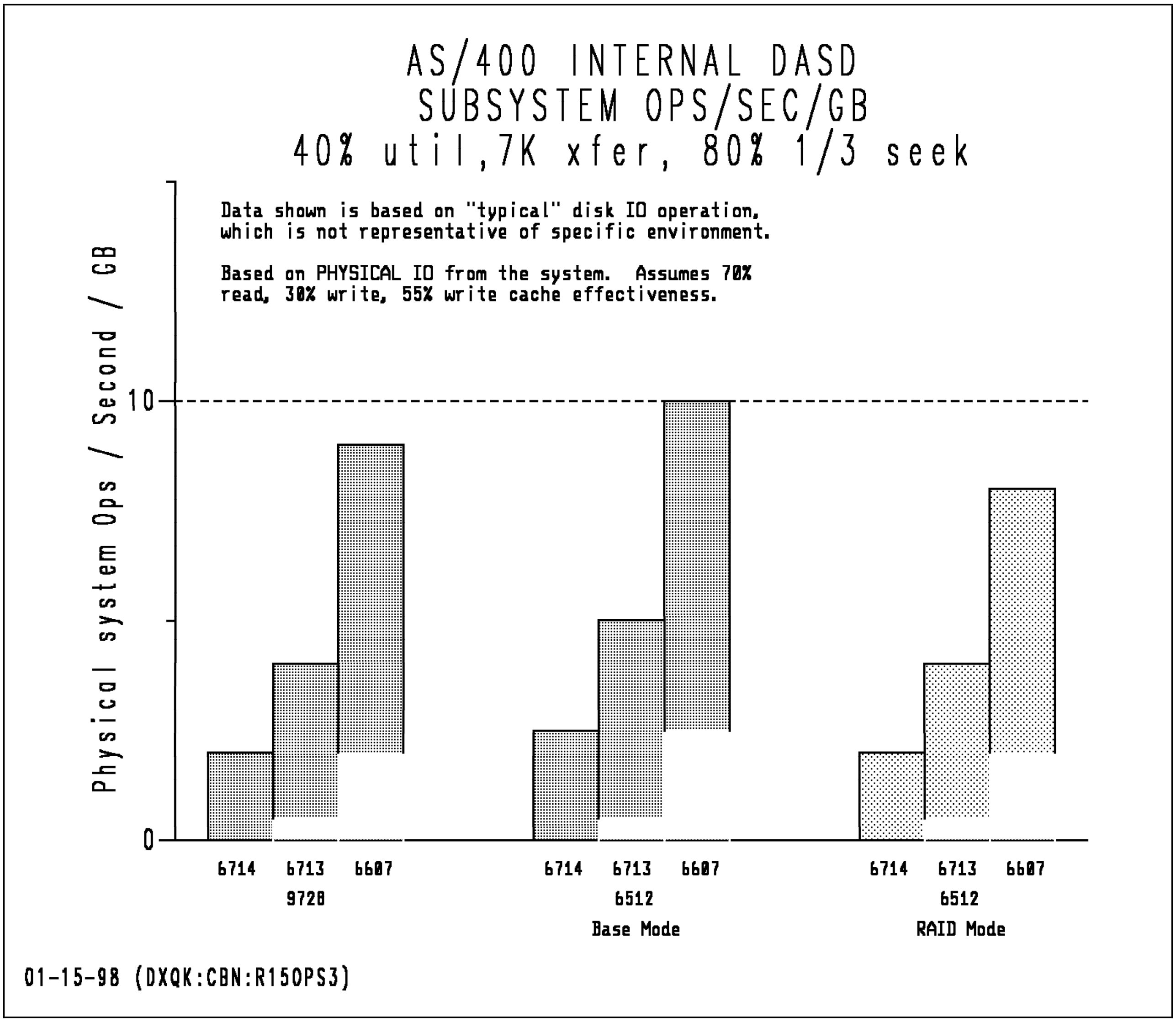


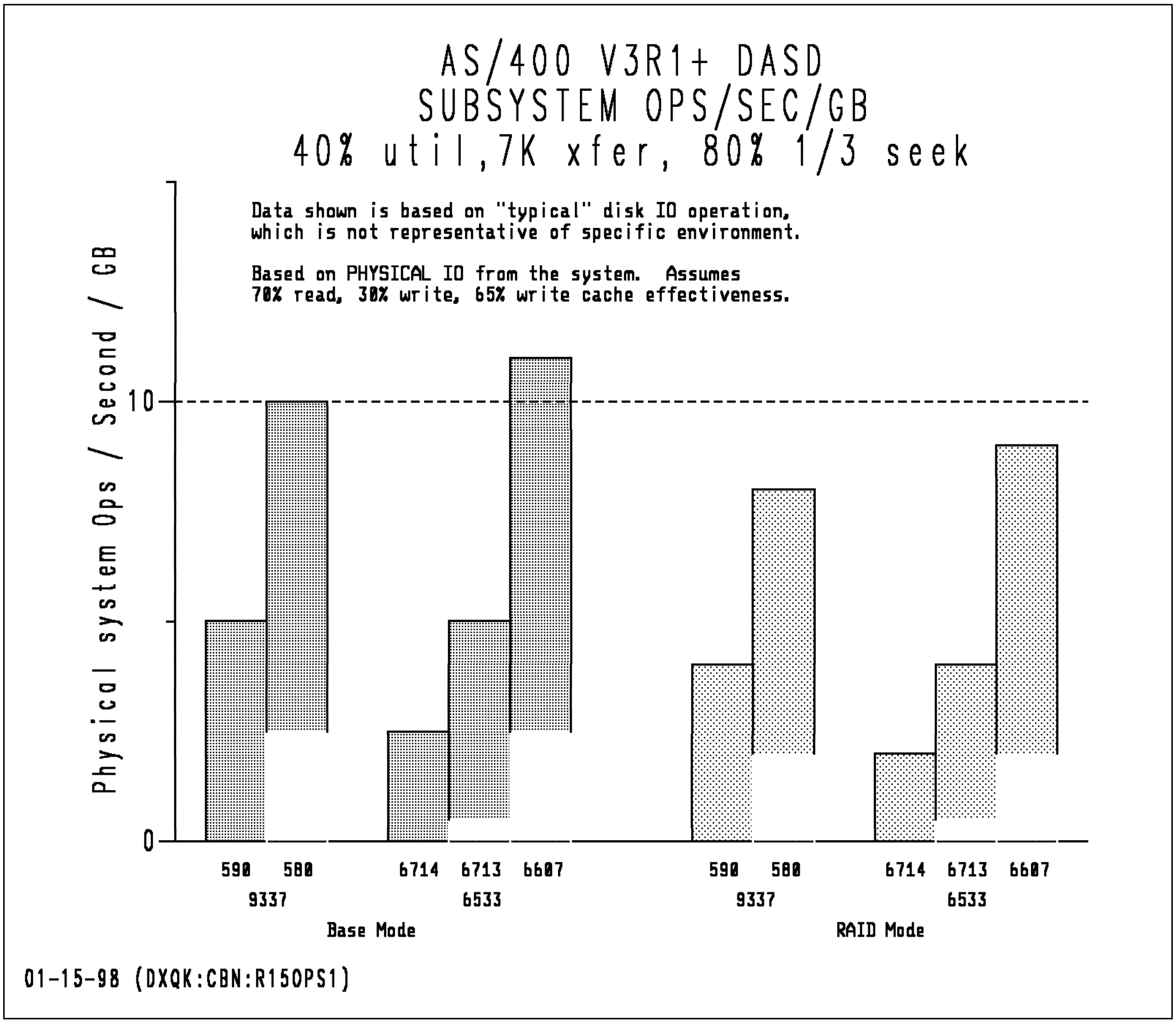
Figure 45. Ops/Sec/GB - Internal DASD (continued)

Conclusions / Recommendations

- In general, the higher the capacity of the DASD device the lower it′s throughput range will be. The 6714 device (17548 MB per arm) has a lower throughput range than the 6606 device (1967 MB per arm).
- Typically, DASD subsystems will have a lower throughput range when operated in RAID mode rather than Base mode. The throughput difference due to RAID will tend to be smaller for workloads characterized by higher read to write ratios.
- The 6714 DASD (17548 MB per arm) is more appropriate when the capacity requirement is very large and the Ops/Sec/GB requirement is less than 3.
- The 6713 DASD (8589 MB per arm) is more appropriate when the capacity requirement is large and the Ops/Sec/GB requirement is less than 4.
- The 6607 models (4194 MB per arm) will be the appropriate choice for almost all other situations. ***NOTE:*** The 6606 models (1976 MB per arm) may be needed for extreme cases where the capacity requirement is very low

relative to the system disk ops/sec rate. Refer to section "Ops/Sec/GB Guidelines for DASD Subsystems" on page 252 for a discussion of the performance limits for each DASD type. 6606 is only needed for cases exceeding the limits for 6607.

- The low cost 9728 DASD IOA provides a similar throughput range when compared to the 6530 DASD IOP.
- The 6512 DASD subsystem provides an improved throughput operating range when compared with the 6502 DASD subsystem.



Data shown is based on "typical" disk IO operation, which is not representative of specific environment.

Based on PHYSICAL IO from the system. Assumes 70% read, 30% write, 65% write cache effectiveness.

01-15-98 (DXQK:CBN:R150PS1)

Figure 46. Ops/Sec/GB - Internal versus External DASD

Conclusions / Recommendations

- The 6533 DASD subsystem provides an improved throughput operating range when compared with the 6512 DASD subsystem. The 6754, 6751, 6532, 2726, 2740 and 2741 DASD IOP/IOAs have performance characteristics similar to the 6533 IOP over typical operating ranges.
- The 6533 DASD subsystem provides an improved throughput operating range when compared with the 9337-5xx DASD subsystem.

- The 6533 DASD IOP provides a better throughput range when compared to the low cost 9728 DASD IOA.

Using the OPS/SEC/GB Chart

The Ops/Sec/GB chart above should be used as a guideline on what DASD model is the appropriate choice when adding or upgrading DASD. In conjunction with this chart, you should utilize results obtained from the Performance Tools LPP report to determine what DASD model meets your DASD performance requirements. For more detailed DASD performance analysis, it is recommended to use BEST/1-400, the capacity planner for the AS/400.

Operations per second is a measurement of throughput per actuator. Since DASD devices have different capacities per actuator, operations per second per GB is used to normalize throughput for different capacities. To determine the operations per second of your current operating environment, follow the procedure outlined below. The value obtained by this procedure will help determine what DASD model will meet your current or projected DASD performance requirements.

1. Collect performance data using the Performance Monitor. Be sure to collect this data during peak activity for at least a one hour time period using 10 minute sample intervals.
2. Print the Performance Tools System Report using the PRTSYSRPT command. Then, refer to the ″Disk Utilization″ section of the Performance Tools LPP System Report. From this report the following data can be obtained:
   - Total operations per second - Use Op Per Second column
   - Total GBs of DASD installed - use Size (M) column
3. To determine the total GBs installed, simply add the ″Size (M)″ column and divide by 1000. When adding the total GBs, you should ONLY include the disk units you plan to replace. Also, if Mirroring is active, divide the total GB being mirrored by 2 when calculating the sum.
4. To determine the total operations per second, add the total operations per second number (″Op Per Second″ column). When adding the total operations per second, you should ONLY include the disk units you plan to replace. Also, if mirroring is active, you need to divide the total number of operations per second for all mirrored units by 2.
5. To determine the operations per second per GB, divide the total operations per second you calculated in step 4, by the total GBs installed value you calculated in step 3.

You can then use the operations per second value to determine what model of DASD best fits your current or projected DASD performance requirement.

EXAMPLE

If you wanted RAID mode, you could select 6533/6607 models if the physical system IO/Sec/GB were at 11 or fewer.

14.3 DASD Performance - Batch

Commercial Batch - Base versus RAID

This section shows the results of running one of IBM′s batch workloads in a dedicated environment. The workload performs the following functions :

- Sequential and Keyed Record Copy
- Sequential and Keyed Program Read
- Sequential and Keyed Record Read/Update
- Record Matching
- Adding and Removing Members
- RGZPFM of 500,000 Records
- Average - 40% Read Ops, 60% Write Ops, 17 KB/IO
- 70% Synchronous / 30% Asynchronous Ops

The workload was run in a 24 MB memory pool. A 6533-18GB was compared to an equivalently configured 6532-18GB with RAID protection turned on and off.

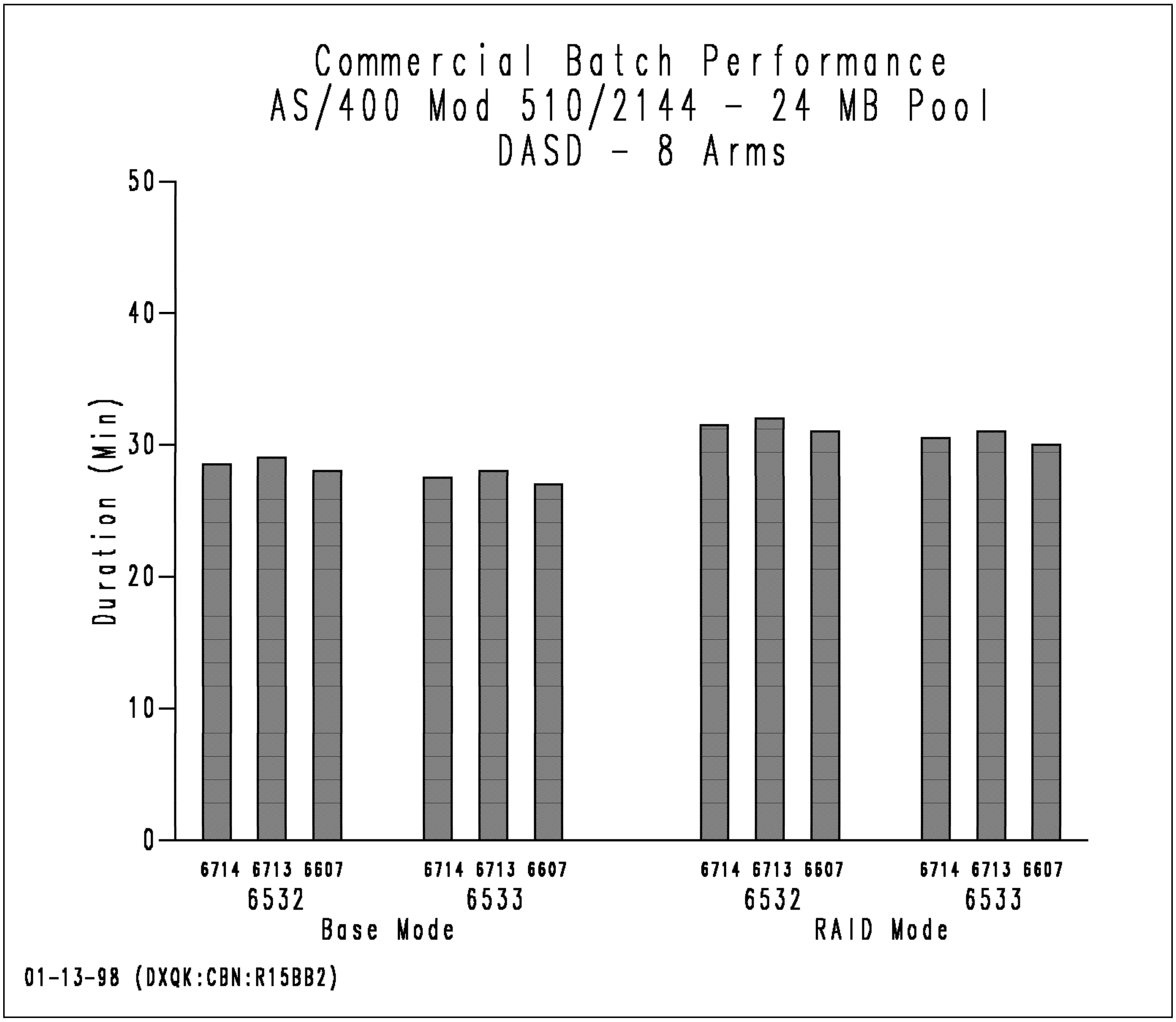


Figure 47. Commercial Batch Performance - Base versus RAID

Conclusions / Recommendations

- The 6533-18GB (6714) has better performance than the 6532-18GB (6614) for this commercial batch workload.
- The 6714 DASD has slightly better performance than the 6713 DASD.

The workload was run in a 24 MB memory pool. A 6512-8GB was compared to an equivalently configured 9337-590 with RAID protection turned on and off.

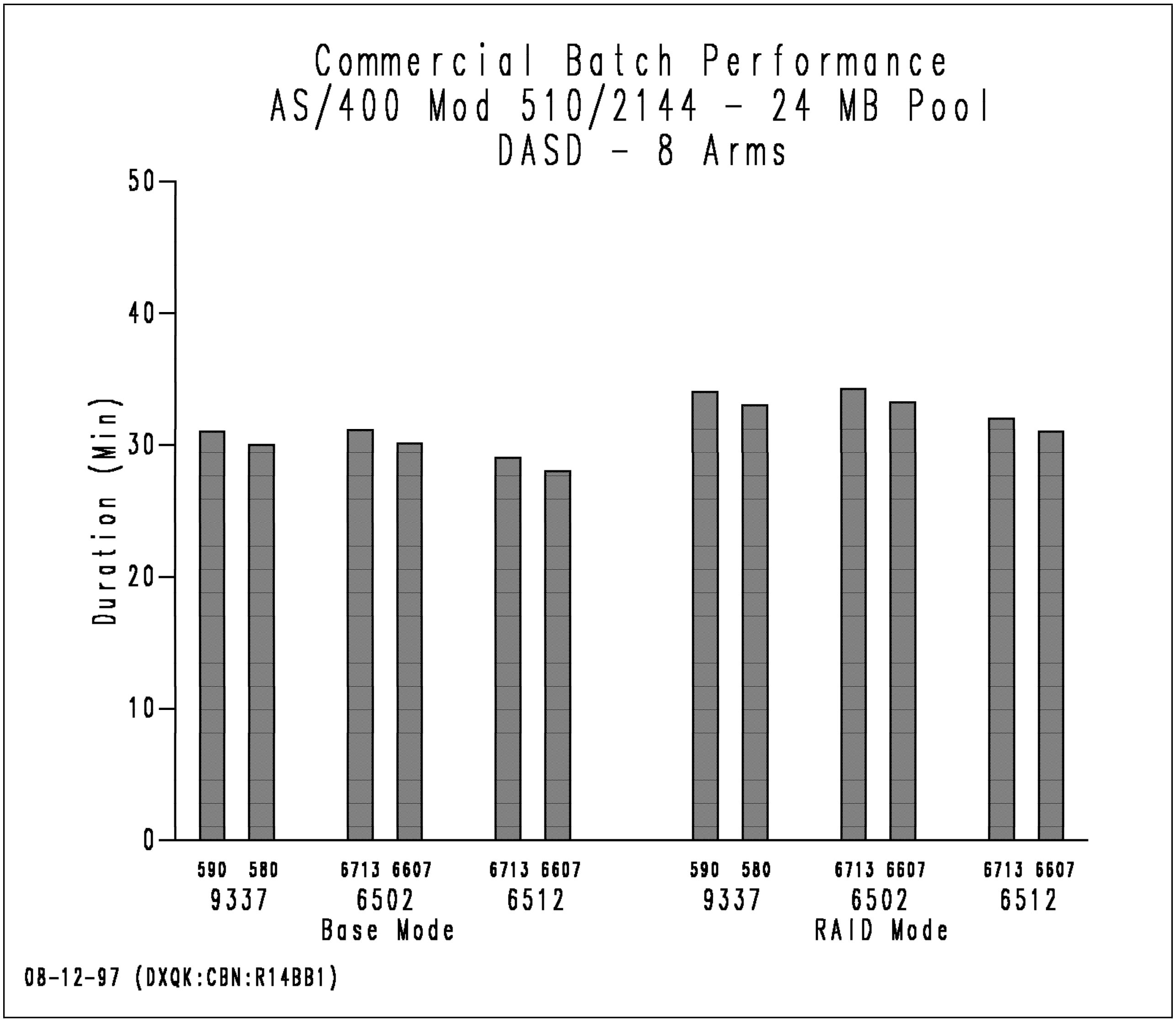


Figure 48. Commercial Batch Performance - Base versus RAID

Conclusions / Recommendations

- The 6512-8GB (6613) has better performance than the 6502-8GB (6613) for this commercial batch workload.
- The 6512-8GB (6613) has better performance than the 9337-590.
- The 6502-4GB (6607) has similar performance as the 9337-580 for this commercial batch workload.
- The 6607 DASD has slightly better performance than the 6713 DASD.

14.4 DASD Performance - General

Mixing RAID DASD with other DASD in one ASP

Combining 65x2 (also 2726, 2740, 2741, 6533, 6751 and 6754) RAID DASD with mirrored DASD in a single ASP is allowed. Combining RAID DASD with mirrored DASD on the same 65x2 is also allowed.

Write Intensive Applications (eg RESTORE)

RAID-5 (like system checksum) can have a significant impact on batch type programs that issue many writes in a short period of time. This is due to the four times increase in disk IO required for each write. The 65x2 write cache handles this impact for almost all scenarios except those that write hundreds of writes to the DASD in a very short period of time. Even in this worst case scenario, with only one 65x2 array configured, the restore of a large file took only 30% longer than a restore to a standard″ 6530 (or 9728) configuration.

The 65x2 can restore small objects faster than 6530 because of the 65x2 write cache. The write cache provides fast completion of write requests and is able to ″stay ahead″ of the system.

The 65x2 RAID models offer significant advantages in availability, reliability, price, etc. One of the ″costs″ of the availability advantage is the increased time to restore data. This increase in time needs to be considered when planning the installation of RAID-5 disk. The time to load data onto the RAID-5 boxes must be included in the overall installation planning. With the increased availability and reliability offered by 65x2 RAID, the necessity to reload the data again due to a single disk unit failure will be eliminated.

DST ″Add Unit″

Part of the process of adding DASD to a system is using the Dedicated Service Tool (DST) to ″Add Unit″. This ensures that the entire DASD(s) is initialized with a X′00′ data pattern and verified.

When multiple DASD are added at once, the system will add up to 16 units in parallel.

The time for adding up to 16 units on a DASD IOP (Base mode) is approximately

- 48 minutes for 2 GB arms (6606)
- 86 minutes for 4 GB arms (6607)
- 162 minutes for 9 GB arms (6713)
- 302 minutes for 18 GB arms (6714)

The Dedicated Service Tool is also used to start and stop parity (RAID-5) arrays on the 65x2 (also 2726, 2740, 2741, 6751 and 6754) IOP. When a parity array is initially set up, the fastest approach is to start parity on an array first and then add the arms to an ASP. The time required for this process (start parity and add) on two 8 arm arrays is approximately

- 48 minutes for 2 GB arms (6606)
- 86 minutes for 4 GB arms (6607)
- 162 minutes for 9 GB arms (6713)

- 302 minutes for 18 GB arms (6714)

If the arms are added to the ASP before starting the array, then the time required may double. If a system IPL occurs between starting the array and adding the arms to an ASP, then the time required could be 3 times as long.

If the arms are currently part of an ASP, then starting an array will take longer because the system may need to move data before it synchronizes the parity stripes. This could take up to

- 90 minutes for 2 GB arms (6606)
- 160 minutes for 4 GB arms (6607)
- 300 minutes for 9 GB arms (6713)
- 580 minutes for 18 GB arms (6714)

Stopping parity on an 8 arm array takes about

- 70 seconds for 2 GB arms (6606)
- 120 seconds for 4 GB arms (6607)
- 220 seconds for 9 GB arms (6713)
- 410 seconds for 18 GB arms (6714)

14.5 Integrated Hardware Disk Compression (IHDC)

Integrated Hardware Disk Compression (IHDC) is a new DASD capability for V4R3. IHDC has the following characteristics :

- Data is dynamically compressed/decompressed by the DASD subsystem controller (IOP/IOA) independent of the AS/400 system processor
- Compressed data is not seen above the DASD controller level
- Compression is performed by an LZ1 compression chip on the DASD controller
- An average 2X compression ratio, with up to 4X achievable (data dependent)
- Compressed DASD performance will typically be slower than uncompressed DASD performance
- Customer on/off option provided at disk arm level
- RAID and mirroring is supported (no additional restrictions)
- With compression, AS/400 disk capacity maximums can be exceeded.

IHDC provides the following customer value :

- Reduces cost of on-line storage
- Provides better performance than typical software data compression
- Enables new applications
- Protects investment - ability to increase capacity of installed DASD
- Provides a storage management solution when used with hierarchical storage management (HSM).

IHDC has the following restrictions :

- Requires a compression enabled IOP/IOA

- DASD compression is limited to user ASPs
- 17.54GB disks not currently supported (will be supported in future)
- Compression can be enabled for unconfigured disks only
- Data must be saved before compression is disabled or it will be lost
- Compressed disks can only be migrated to a system that has compression enabled IOP/IOAs.

Interactive Performance with Compressed DASD

The following graph compares relative interactive system performance of an AS/400 model 640/2239 configured with 16-arm user ASPs of Compressed, Uncompressed, RAID/Compressed and RAID/Uncompressed DASD. The graph compares the performance results of running the RAMP-C workload in each of the 4 user ASPs.

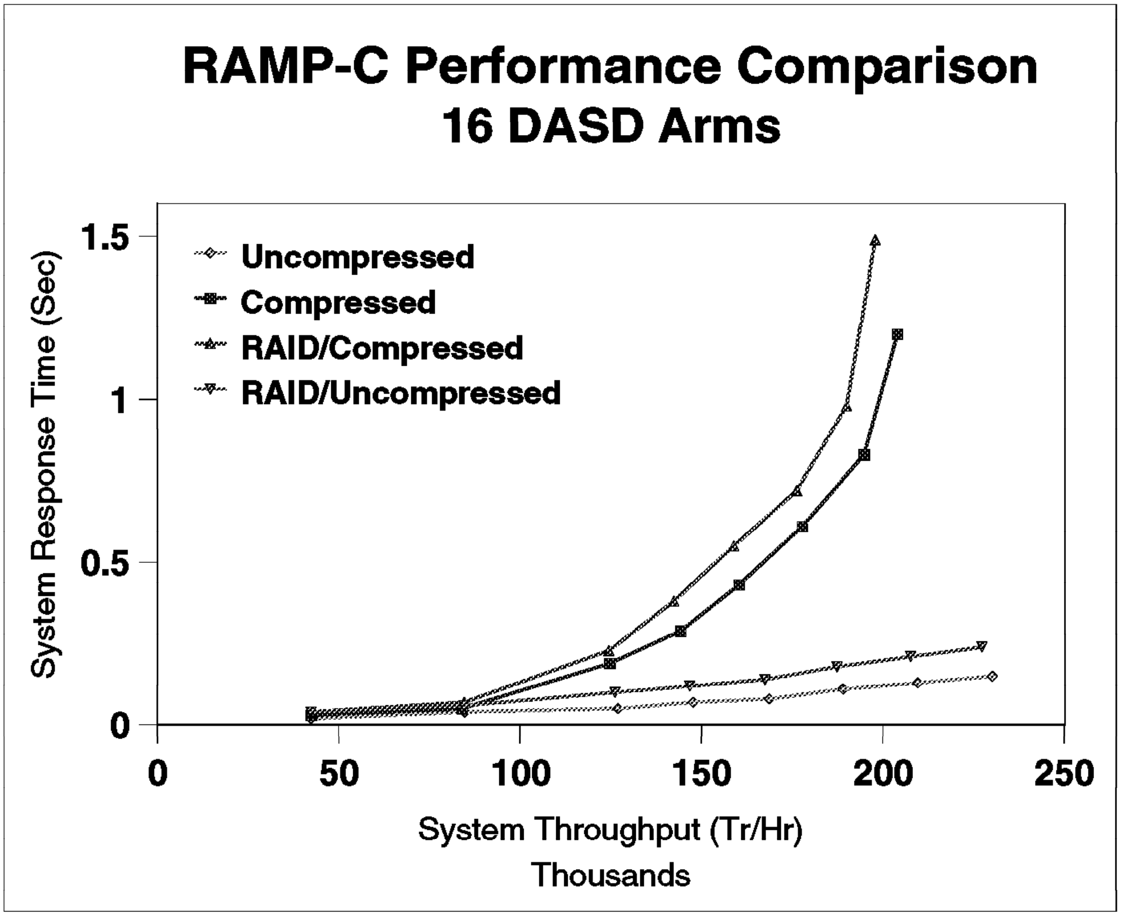


Figure 49. System Interactive Performance - Compressed DASD

Conclusions / Recommendations

- At lower throughputs and with equal number of disk arms, Compressed DASD has similar (0-10% degradation) system performance characteristics to Uncompressed DASD for interactive workloads. However, if workloads are write intensive or DASD throughputs are higher, Compressed DASD performance could be much worse (10-50% degradation).
- Acceptable Compressed DASD performance can usually be obtained by maintaining :
  - Disk CPU utilization below 60% (about 360 ops/sec per IOP/IOA)
  - Disk utilization below 40% (about 23 ops/sec per disk arm)
- For the above graph, the Compressed disk 40% utilization point is at 146,000 transactions per hour and the RAID/Compressed disk 40% utilization point is at 140,000 transactions per hour.
- For the above graph, the Compressed Disk CPU (IOP/IOA) 60% utilization point is at 144,000 transactions per hour and the RAID/Compressed Disk CPU 60% utilization point is at 138,000 transactions per hour.
- Above these limits, system performance with Compressed DASD tends to degrade noticeably (10-90%) as the throughput increases.
- Compressed DASD with RAID has similar system performance characteristics to Compressed DASD without RAID at lower throughputs. At higher throughputs, RAID/Compressed DASD performance is less due to higher utilization′s for the same op rates. The same criteria as above should be followed for obtaining acceptable RAID/Compressed DASD performance.
- For equal DASD capacity, Compressed DASD has the potential of performing much worse than Uncompressed DASD due to the fewer disk arms required (approximately one half). Care must be taken to ensure that enough disk arms are configured to support the system op rate anticipated.
- Configuring fewer disk arms per IOP/IOA will typically improve the performance of Compressed DASD. DASD subsystems with 8 arms per IOP/IOA will usually perform much better than those with 16 arms per IOP/IOA.

Batch Performance with Compressed DASD

The following chart compares system performance of various batch type applications while running on an AS/400 model S30/2259 configured with 16-arm user ASPs of Compressed, Uncompressed, RAID/Compressed and RAID/Uncompressed DASD. Batch run time was measured in each of the 4 user ASPs for 7 batch tests with the following DASD I/O characteristics :

1. Sequential read ops, 5 KB/op, OS/400 Expert Cache off
2. Sequential read ops, 60 KB/op, OS/400 Expert Cache on
3. Sequential read and write ops, 68% reads, 5 KB/op, OS/400 Expert Cache off
4. Sequential read and write ops, 17% reads, 50 KB/read op, 5 KB/write op, OS/400 Expert Cache on
5. Random read ops, 7 KB/op, OS/400 Expert Cache off
6. Random write ops, 8 KB/op, OS/400 Expert Cache off
7. Sequential read and write ops, 14% reads, 5 KB/op, OS/400 Expert Cache off

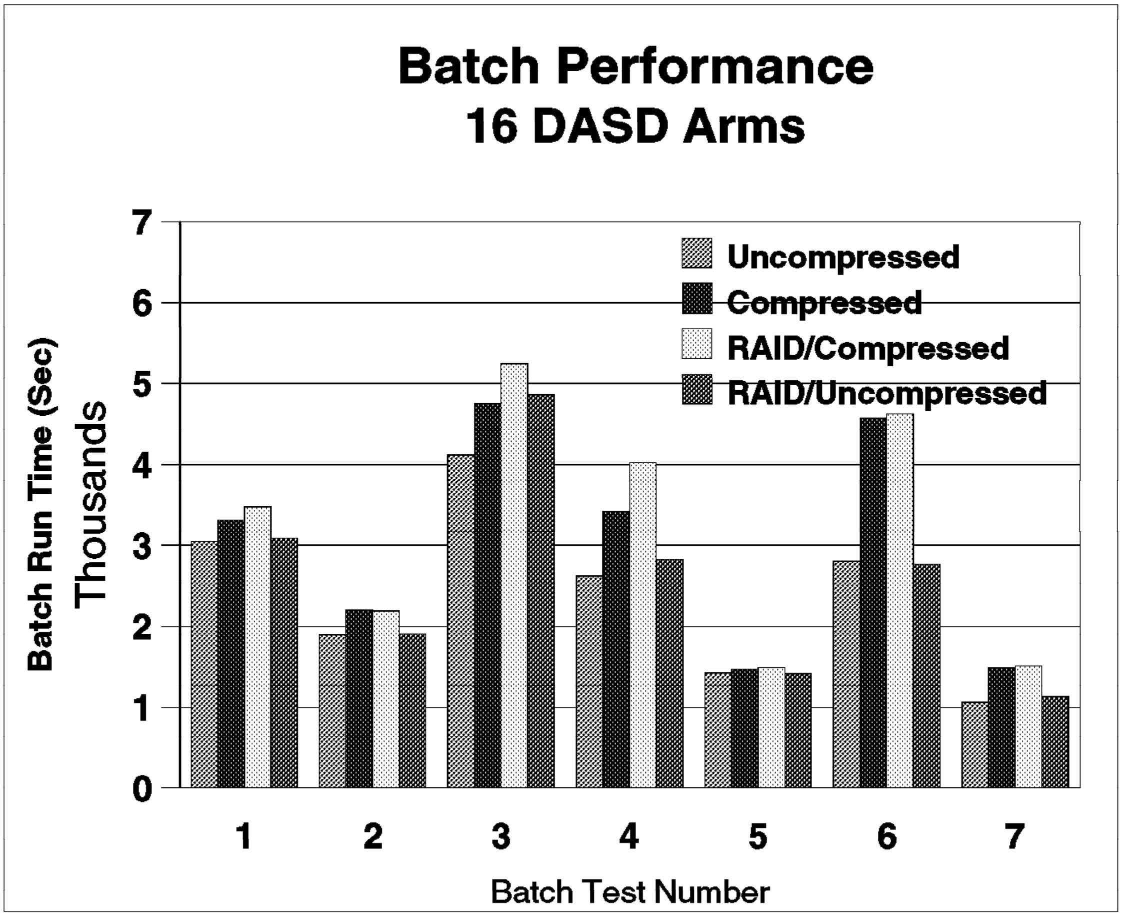


Figure 50. Batch Run Time Performance - Compressed DASD

Conclusions / Recommendations

- For batch applications characterized by DASD read ops, system performance varied only slightly between Compressed, Uncompressed, RAID/Compressed and RAID/Uncompressed DASD ASPs.
- For batch applications characterized by DASD write ops, system performance was slower for Compressed and RAID/Compressed than for Uncompressed and RAID/Uncompressed DASD ASPs.
- For batch applications characterized by a mixture of DASD read and write ops, system performance was slower for Compressed and RAID/Compressed than for Uncompressed and RAID/Uncompressed DASD ASPs. The magnitude of the performance difference typically depends on the percentage of write ops; the larger the percentage of write ops, the slower the performance.
- OS/400 Expert Cache provided better batch system performance when active.

Save/Restore Performance with Compressed DASD

The following charts compare system performance of Save/Restore operations while running on an AS/400 model S30/2259 configured with 16-arm user ASPs of Compressed, Uncompressed, RAID/Compressed and RAID/Uncompressed DASD. The System ASP (ASP1) was configured with Uncompressed DASD. Data transfer rates were measured in each of the 4 user ASPs for 6 different types of Save/Restore tests :

1. SAV to a 3590 tape from the user ASP (Read data from Compressed DASD)
2. RST from a 3590 tape to the user ASP (Write data to Compressed DASD)
3. SAV from ASP1 to a *SAVF on the user ASP (Write data to Compressed DASD)
4. RST from a *SAVF on the user ASP to ASP1 (Read data from Compressed DASD)
5. SAV to a *SAVF on ASP1 from the user ASP (Read data from Compressed DASD)
6. RST from a *SAVF on ASP1 to the user ASP (Write data to Compressed DASD)

The data used for the first chart has a compression ratio of 2X and the data for the second chart has a compression ratio of 4X.

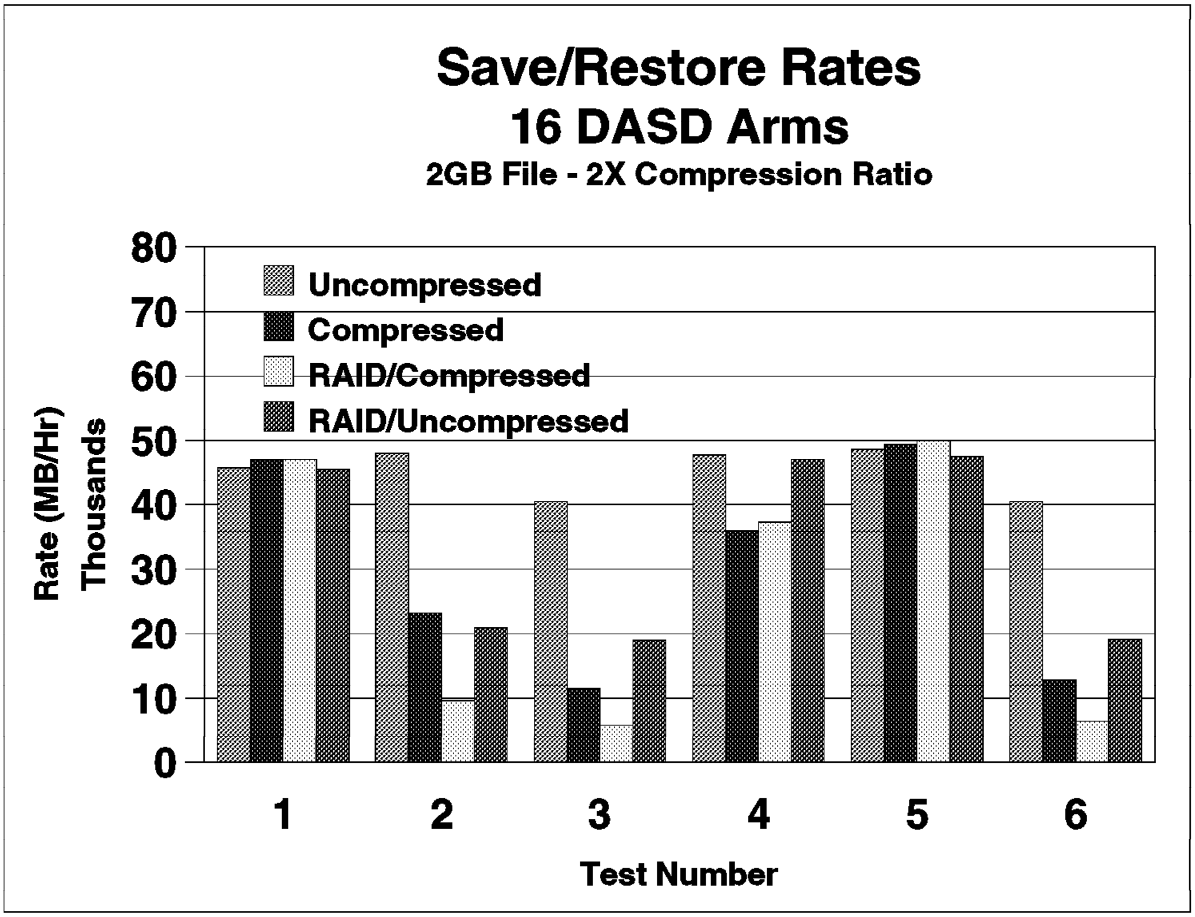


Figure 51. Compressed DASD Save/Restore Rates - 2X Compressibility

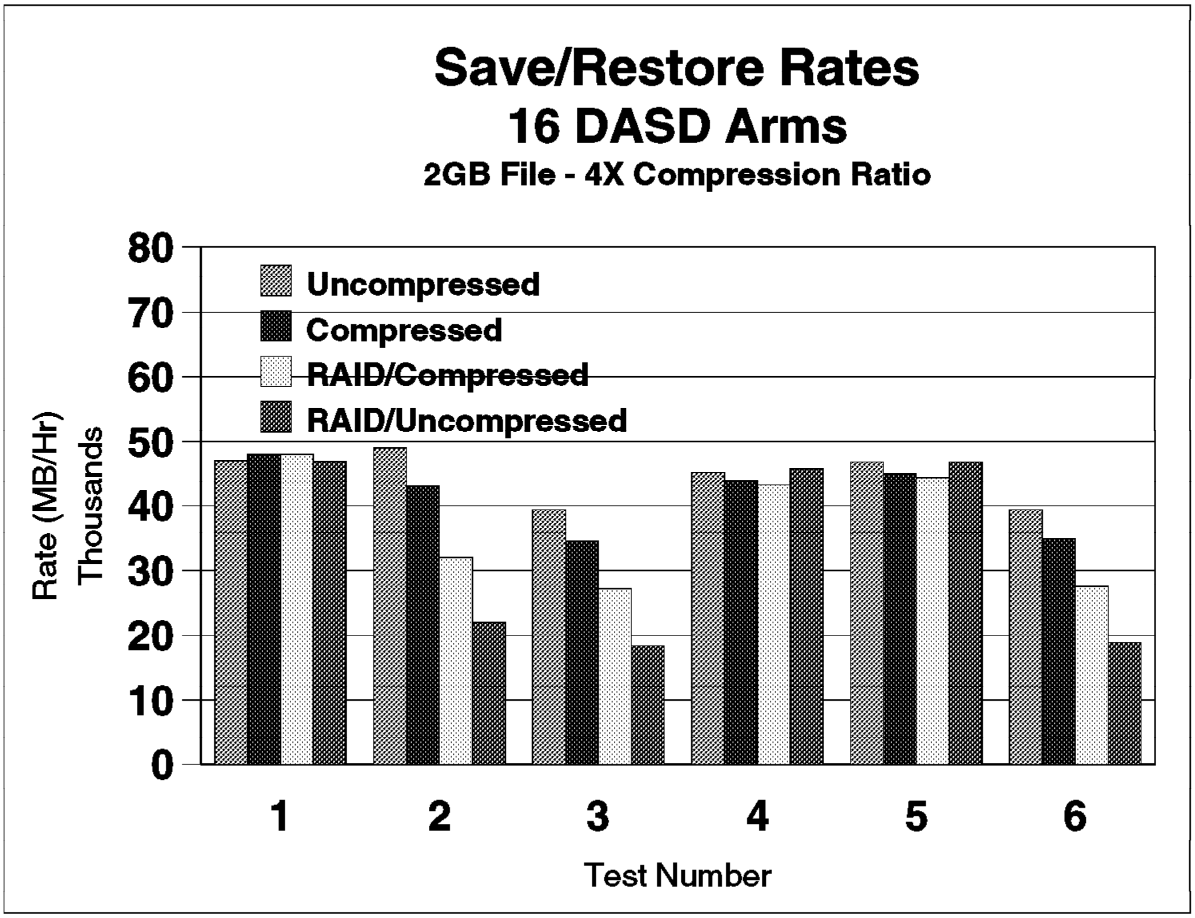


Figure 52. Compressed DASD Save/Restore Rates - 4X Compressibility

Conclusions / Recommendations

- For Save operations to tape, Compressed DASD (also RAID/Compressed DASD) has approximately the same system performance characteristics as Uncompressed DASD. Save operations primarily issue read op to DASD and read op performance is very similar for Compressed, RAID/Compressed and Uncompressed DASD.
- For Restore operations from tape, Compressed DASD performance is highly dependent upon the compressibility of the data being restored - the better the data compresses the better the restore performance.
  - Performance can range from 50% degradation (2x compression) to 10% degradation (4x compression) for Compressed DASD compared to Uncompressed DASD.
  - Performance can range from 50% degradation (2x compression) to 40% improvement (4x compression) for RAID/Compressed DASD compared to RAID DASD. The improvement is due to better write cache efficiency and smaller ops because of data compression.
  - Restore operations primarily issue large write ops to DASD along with allocate ops. Performance degradation occurs because ops have higher

overhead due to compression and generate higher utilization for the IOP/IOA and devices.

Data Migration Performance with Compressed DASD

The following chart compare system performance of BRMS data migration operations while running on an AS/400 model S30/2259 configured with 16-arm user ASPs of Compressed and Uncompressed DASD. The System ASP (ASP1) was configured with Uncompressed DASD. Data transfer rates were measured between ASP1 and each of the 2 user ASPs for 4 different types of data :

- Large data base file
- Typical user mix of data
- Source data
- DLO data

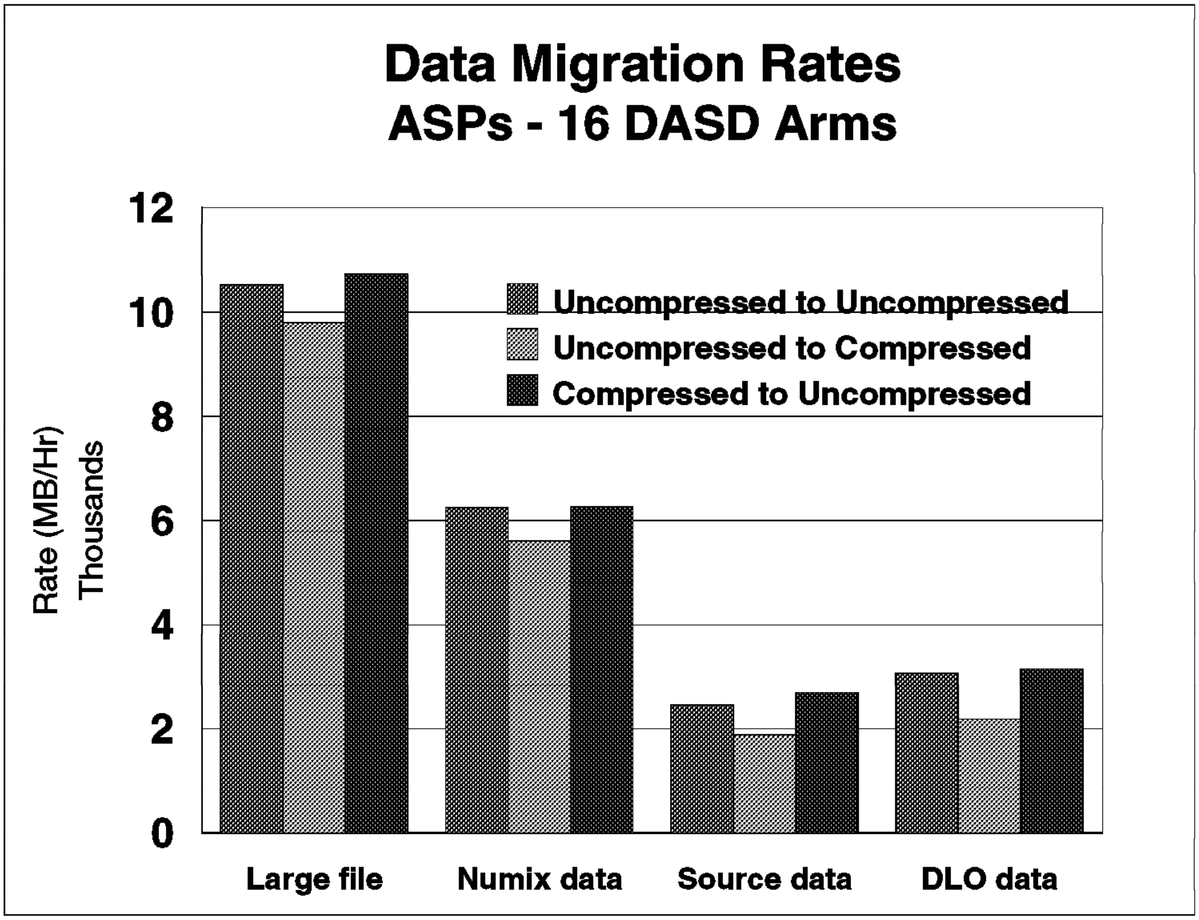


Figure 53. Compressed DASD Data Migration Rates

Conclusions / Recommendations

- Migrating data from an Uncompressed ASP to a Compressed ASP is usually slower than migrating data between two Uncompressed ASPs. The magnitude of the performance difference typically depends on the type and compressibility of the data being migrated, the lower the compressibility, the slower the transfer rate.
- Migrating data from a Compressed ASP to an Uncompressed ASP is usually as fast as or slightly faster than migrating data between two Uncompressed ASPs.

Data Throughput Performance with Compressed DASD

The following graph compares relative data throughput performance of Compressed and Uncompressed DASD read and write operations for the range of data compression ratios supported. This graph is intended to show in general how data throughput for Compressed DASD is dependent upon the compressibility of the data being transferred.

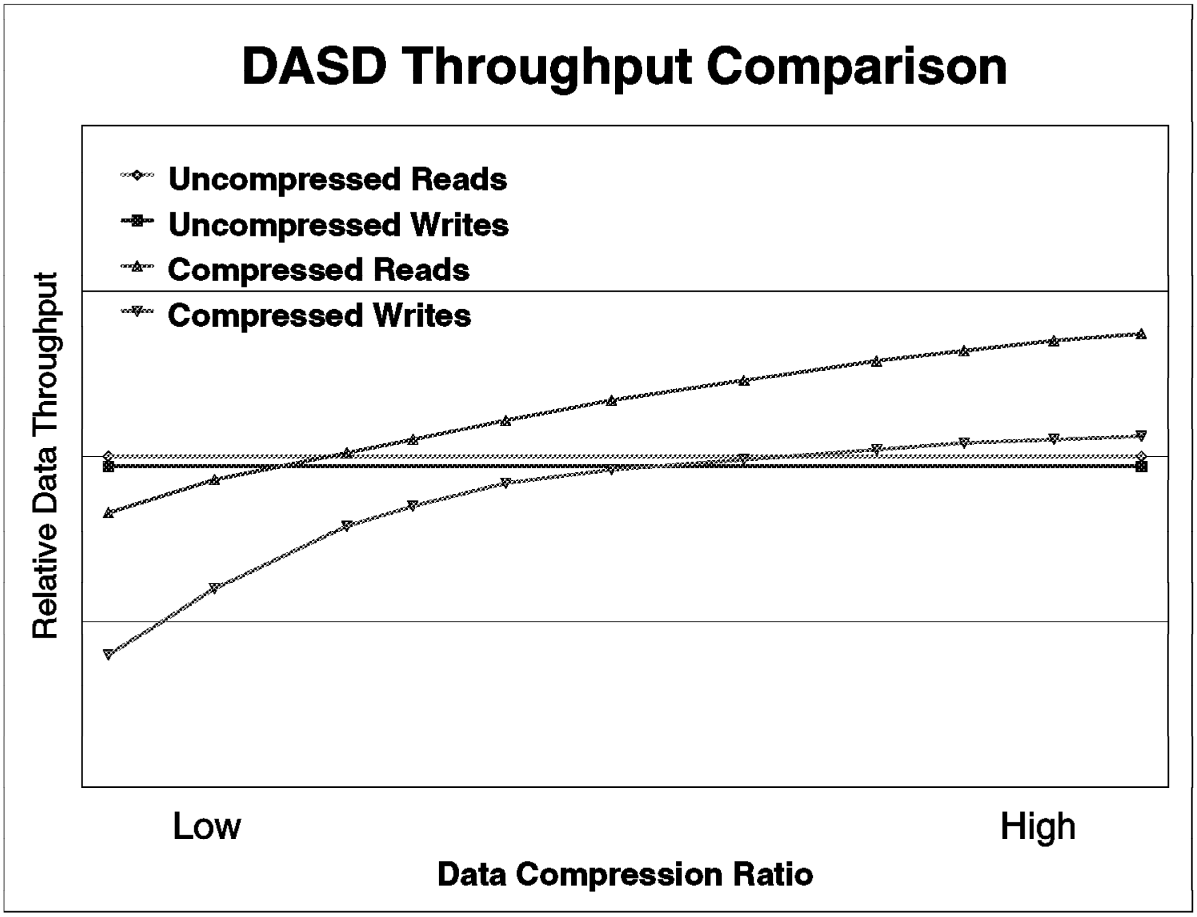


Figure 54. Compressed DASD Relative Data Throughput

Conclusions / Recommendations

- Data throughput for Uncompressed DASD read and write operations is independent of the compression ratio of the data.
- Data throughput for read operations from Compressed DASD is slightly less than from Uncompressed DASD at low compression ratios but can be much better at higher compressions ratios.
- Data throughput for write operations to Compressed DASD is much less than to Uncompressed DASD at low compression ratios but can be better at higher compressions ratios.

DASD Compression Performance Guidelines

Precise performance projections for IHDC are not possible due to :

- Compressibility of data which can vary greatly
- Workload characteristics of specific applications

DASD compression may cause performance to vary, in general :

- For data with low compression rates (2-3X), DASD read/write performance will generally be slower than for uncompressed data
- For data with high compression rates (3-4X), DASD read/write performance can be faster than for uncompressed data
- DASD Read intensive workloads will typically perform better than DASD Write intensive workloads
- System performance impacts will be minimal when DASD operations are light
- Interactive applications with a mixture of DASD reads and writes with medium to heavy DASD operations should use high performance uncompressed DASD
- With compressed disks it is critical that they operate within a reasonable margin below ′dasd full′, otherwise performance will be greatly affected. In contrast to uncompressed disks, when a compressed disk approaches full (approximately 85%) a disk defragmentation task is started within the IOP/IOA to recover fragmented storage and may take considerable time to finish. Since this task runs concurrently with system operations, performance will be degraded until this task completes.
- Mixing Compressed DASD with Uncompressed DASD on the same IOP/IOA may impact the performance of the Uncompressed DASD due to higher IOP/IOA utilization.
- Mixing Compressed DASD with Uncompressed DASD within the same user ASP is supported but is not recommended due to potential performance impact caused by unbalanced disk utilization.

DASD compression is intended for :

- Vast amounts of historical or archive data
- Low activity data
- Spool files
- Journals

- Save files (staging)

Not for highly volatile data or data already compressed (images, etc.)

Types of applications that can benefit from DASD compression :

- Data warehouse, data mining
- On-line access to archive data
- On-line viewing of reports (paper or micro-fiche replacement)
- Part of a hierarchical storage management strategy

Applications are a candidate for DASD compression if :

- Additional DASD storage is required
- Application data can be partitioned (at least partially) into user ASPs
- Top application performance is not required

Refer to *AS/400 Backup and Recovery V4R3* (SC41-5304-02) for more information about configuring and using compressed DASD.

Chapter 15. Save/Restore Performance

Many factors influence the observable performance differences of save operations and restore operations. These factors include:

- Hardware (such as tape drives)
- The tape Input/Output Processor (IOP) that is used
- Placement of the tape IOP in the system
- Type of work load (Large file, User Mix)

The use of data compression, data compaction, and the Use Optimum Block Size (USEOPTBLK) parameter also influence the performance of save operations and restore operations.

As you look at tape drives and their performance rates, you need to understand the tape hardware and the capabilities of that hardware. The different tape drives and IOPs have different capabilities to manipulate data for the best results in their target market. With this in mind, it should be noted that a slower rated tape drive can actually be faster for some customer environments. The following table shows the tape drives that were used in the work load tests and the rates for each drive. (The rates are used later in this document to determine overall performance.)

A study of the data compaction was performed on a general sampling of customer data. The study found that compression occurred at a ratio of aproximately 2.8 to 1. The performance data for the 2GB work loads is based on this ratio for tape drives that use LZ1 for this compaction algorithm, and the same data compacts at about 1.8 to 1 for IDRC drives.

Drive	Rate (MB/S)	COMPACT	USEOPTBLK
6380	0.3	0.0	
6382/6383	0.38	0.0	
6390	0.5	1.8	
7208-342	3.0	1.8	YES
6385	1.5	1.8	
3570	2.2	2.5	YES
3570-C	5.5	2.5	YES
3590	9.0	1.6[1]	YES

[1] The 3590 does not make use of full compaction. The figures listed here are an attempt to simulate the way it actually works today.

15.1 Use Optimum Block Size (USEOPTBLK) parameter

The USEOPTBLK parameter is used to send a larger block of data to tape drives that can take advantage of the larger block size. Every block of data that is sent has a certain amount of overhead that goes with it. This overhead includes block transfer time, IOP overhead, and drive overhead. The block size does not change the IOP overhead and drive overhead, but the number of blocks does. For example, sending 8 small blocks will result in 8 times as much IOP overhead and Drive overhead. If the same amount of data is sent in 1 large block, the result is

1 times as much overhead. With the larger block size, the IOP overhead and Drive overhead become less significant. This allows the actual transfer time of the data to become the gating factor. In this example, 8 software operations with 8 hardware operations essentially become 1 software operation with 1 hardware operation when USEOPTBLK(*YES) is specified. The usual result is significantly lower CPU utilization. This also allows the tape device to perform more efficiently.

15.2 Data Compression (DTACPR)

Data compression is the ability to compress strings of identical characters and mark the beginning of the compressed string with a control byte. Strings of blanks from 2 to 63 bytes are compressed to a single byte. Strings of identical characters between 3 and 63 bytes are compressed to 2 bytes. If a string cannot be compressed a control character is still added which will actually expand the data. This parameter is usually used to conserve storage media.

If the IOP does not support data compression, the software performs the compression. This situation can require a considerable amount of processing power. One of the exceptions to this is the 2644 IOP card for the 3490 tape drive. In this situation, the IOP is designed to use compression as noted in 15.12, Save/Restore Tips for Better Performance on page 280 of this document.

15.3 Data Compaction (COMPACT)

Data compaction is only available at the hardware level. If you wish to use data compaction, the tape drive you choose will need to support it.

15.4 Work Loads

The following work loads were designed to help evaluate the performance of save operations and restore operations. Familiarization with the makeup of the work loads will help clarify the differences in the save rates and restore rates that are presented below.

NUMX The User Environment work load (NUMX) consists of 4 libraries. The first library contains 4 source files (for a total of 1204 members) that comprise about 39 MB of space. The second library consists of 28 database files, ranging in size from 2 MB to 200 MB, which total 470 MB in size. The third library consists of 200 program objects, with an average size of about 100 KB, for a total size of 20 MB. The fourth library is 12 MB in size and consists of 2156 objects of various types. Therefore, the NUMX work load consists of about 556 MB.

NSRC The NSRC work load consists of the 4 source files that are in the first library of the NUMX. These source files occupy about 39 MB of space and contain a total of 1204 members.

2GB The 2GB work load is a single member database file that is about 2 GB in size. This work load is saved using the SAVOBJ command and restored using the RSTOBJ command.

DLO The DLO work load consists of 8 folders with 3700 documents in the folders. The documents range in size from 53K to 233K with a combined size of 396MB. All of the documents reside in the first level folder structure.

Integrated File System The type of data stored in the integrated file system has been changing since it was made available in OS/400 V3R1M0. In the past, the type of data stored in the file system mainly consisted of client programs. Programs don′t compact or compress so they are saved or restored at the basic rate of the tape drive being used. With the introduction of Lotus Notes and Web functions, more files that contain data are being stored in the file system. With these changes, the rate at which the RST and SAV commands complete has changed because these objects can take advantage of the compaction capabilities of the tape drives.

The following describes save and restore rates that a customer might see depending upon their data and its compaction capabilities. Take a system with an even mixture of client programs, Lotus Notes databases and Web home pages. This example should save and restore in the range of the NUMX workload described in our charts. If the data stored on the system is largely made up of database files, the save/restore rates will probably tend toward the 200M- to-large file type of workload, depending on the size and number of database files. If the data is largely made up of Web files, which tend to be numerous small HTML files such as small home pages, the save rates will tend downward from NUMX toward the NSRC workload.

Web objects can be large images and client databases, just as Lotus Notes database files can be numerous empty or near empty mail files. This would reverse the description above. In all situations the actual data will dictate the save/restore rates and the customer will need to know the type of data they have on their systems in order to estimate the save or restore rates.

The **concurrent save/restore** work load is the ability to save/ restore a single library to multiple tape drives at the same time. You can generally expect to save 1.8 times as much data as with one tape drive in the same amount of time. The work loads that were used for this testing were two 2 GB files in the same library, and the NUMX work load. For the NUMX work load, all objects from the 4 NUMX libraries were combined into one library, and all of the objects were then duplicated. The duplicate objects were named in such a way that the original objects would go to one tape drive and the duplicate objects to another. Example: APF200MB and BPF200MB.

Most of the Save/Restore rates were obtained from a restricted state measurement. (A restricted state measurement is performed when all subsystems are ended using the ENDSBS(*ALL) command.) The new work loads for concurrent Saves were on a dedicated system (Dedicated is different than Restricted in that in dedicated subsystem QBATCH is running to allow the concurrent jobs to be running.) All work loads were deleted before restoring them again.

15.5 Comparing Performance Data

When comparing the performance data in this document with the actual performance on your system, remember that the performance of save operations and restore operations is data dependent. If the same save operation or the same restore operation is performed on data from three different systems, three different rates may result, even on the same tape drive. The performance fluctuation is dependent on the data itself.

The performance of save operations and restore operations is also dependent on the configuration of the DASD units on which the data is stored . When more DASD units are installed on the system, performance generally improves.

Generally speaking, the large file data that was used in testing for this document was designed to compact at an approximate 2.6:1 ratio. If we were to write a formula to illustrate how performance ratings are obtained, it would be as follows:

((TapeSpeed * LossFromWorkLoadType) * Compaction) = MB/Sec

We would then multiply MB/Sec * 3600 = MB/HR.

But the reality of this formula is that the LossFromWorkLoadType is far more complex than described here. The different work loads have different overheads, different compaction rates, and the drives use different buffer sizes and different compaction algorithms. The attempt here is to group these work loads as examples of what might happen with a certain type of drive and a certain work load.

Note: Remember that these formulas and charts are to give you an idea of what you might achieve from a particular tape drive. Your data is as unique as your company and the correct tape solution must take into account many different factors. These factors include system size, tape drive model, the number of tapes that are required, and whether you are performing an attended or unattended save operation.

15.6 Lower Performing Tape Drives

With the lower performing tape drives (6380, 6390), the drives themselves become the gating factor, so the save rates are approximately the same, regardless of system size . For testing purposes on the 6380, compression was substituted for compaction in the formula . Therefore, the formula that was used to determine tape speed was (TapeSpeed * LossFromWorkLoadType).

The following are LossFromWorkLoadType approximations (for save operations) to use in the formula when looking at the small tape drives.

Table 80. LossFromWorkLoadType Approximations (Save Operations)

Work Load Type	**Amount of loss**
2GB	95%
NUMX, DLO	85%
NSRC	50%

Compression for the 6380 = 1.8 and compaction for the 6390 = 1.8.

```
        TapeSpeed * LossFromWorkLoadType * Compaction
6390       .5      *      .95=(.475)          *     1.8      =(.713) MB/S * 3600  =  3078 MB/HR
```

Figure 55. 6390 Saving large files — Example

15.7 Medium Performing Tape Drives

The overhead is different for the medium performing tape drives (6385, 3570-B). These tape drives have been designed with different technologies and with different markets in mind. It is the differences in these drives that make them difficult to compare. The 3570-B uses optimum block size and LZ1 compaction. Use of USEOPTBLK(*YES) can make the 3570 an efficient drive for systems that are CPU-constrained.

Table 81. LossFromWorkLoadType Approximations (Save Operations)

Work Load Type	Amount of loss
2GB	85%
NUMX, DLO	65%
NSRC	25%

```
        TapeSpeed * LossFromWorkLoadType * Compaction
6385        1.5      *       .85=(1.28)             *     1.8        = (2.3) MB/S * 3600 = 8280 MB/HR
3570        2.2      *       .85=(1.87)             *     2.5        = (4.68) MB/S * 3600 = 16848 MB/HR
```

Figure 56. 6385 and 3570-B Saving Large Files — Example

15.8 Highest Performing Tape Drives

The overhead for the highest performing tape drives (2570-C and 3590) is also different from the other types of tape drives. The high speed tape drives take advantage of optimum block, and high speed data transfer rates allowing large system users to keep their save windows at a minimum. See 15.12, "Save/Restore Tips for Better Performance" on page 280 for information on getting the most out of your high speed tape drives. The high speed tape drives perform best on large files, as shown in tables 6 through 9.

Table 82. LossFromWorkLoadType Approximations (Save Operations)

Work Load Type	Amount of loss
2 GB	95%
NUMX, DLO	44%
NSRC	12%

Note: The 3590 does not make full use of compaction, and the 1.6 is an attempt to simulate the way it actually works today.

```
        TapeSpeed * LossFromWorkLoadType * Compaction
3570-C        5.5      *        .95=(8.6)              *     2.5        =(13.06) MB/S *3600 = 47000 MB/HR
3590         9.0     *        .95=(8.6)             *     1.6       =(13.76) MB/S *3600 = 49536 MB/HR
```

Figure 57. Saving Large Files — Example

15.9 V4R3 Save and Restore Rates

The save/restore rates that are quoted below are expressed in terms of megabytes per hour (MB/HR). All of these measurements used the options for compression (DTACPR) and compaction (COMPACT) as shown in the section on performance tips on page 280. The USEOPTBLK parameter is always used where it is available and is noted by the * after the tape device.

Note: PLEASE READ THE OTHER SECTIONS TO UNDERSTAND HOW THESE NUMBERS WERE ACHIEVED.

Table 83. Save rates for the Model 400 feature 2133

Tape Device	**Work Load**			
	NSRC	NUMX	2GB	DLO
6380	1310	1800	1890	1780
6390	1350	2500	2600	3190
7208-342*	2340	10200	15500	14990
3570-B*	4550	11550	20000	14000
3570-C*	3300	12600	23900	12200

* = Save performed with USEOPTBLK(*YES)

Table 84. Restore rates for the Model 400 feature 2133

Tape Device	**Work Load**			
	NSRC	NUMX	2GB	DLO
6380	610	1740	1890	1780
6390	850	2000	2600	3190
7208-342[1]	1610	6840	15640	3520
3570-B[1]	2300	8000	19000	3800
3570-C[1]	1590	7200	22400	2990

[1] The save operation must have been performed with USEOPTBLK(*YES) for these rates to be correct.

Table 85. Save rates for the Model 510 feature 2144

Tape Device	**Work Load**			
	NSRC	NUMX	2GB	DLO
6380	1410	1760	1900	1780
6382	1830	3560	4050	3640
6390	1280	2470	2625	3310
7208-342*	2240	9970	15580	14790
6385	1320	5300	9640	1980
3570-B*	4500	11800	20600	12400
3570-C*	4170	18600	35900	20400
3590*	6100	18600	44750	24300

* = Save performed with USEOPTBLK(*YES)

Table 86. Restore rates for the Model 510 feature 2144

Tape Device	Work Load			
	NSRC	NUMX	2GB	DLO
6380	1120	1680	1894	1730
6382	1350	2010	2180	2690
6390	1560	2400	2700	3170
7208-342[1]	1980	8220	15850	4540
6385	1470	2390	9530	2260
3490	2400	11200	14300	4650
3570-B[1]	2800	9000	20500	4700
3570-C[1]	3000	14000	40500	5900
3590[1]	3100	12100	39900	5010

[1] The save operation must have been performed with USEOPTBLK(*YES) for these rates to be correct.

Table 87. Save rates for the Model S20 feature 2166

Tape Device	Work Load			
	NSRC	NUMX	2GB	DLO
3570-B*	5800	14100	20800	15000
3570-C*	4648	21900	41500	20200
3590*	6200	24000	49800	25000

* = Save performed with USEOPTBLK(*YES)

Table 88. Restore rates for the Model S20 feature 2166

Tape Device	Work Load			
	NSRC	NUMX	2GB	DLO
3570-B[1]	4500	14400	21000	10500
3570-C[1]	6000	22800	41000	17800
3590[1]	4690	23900	49900	20000

[1] The save operation must have been performed with USEOPTBLK(*YES) for these rates to be correct.

Table 89. Model S20 feature 2166 Concurrent Saves		
Tape Device	**Work Load**	
	NUMX	2GB
3570-B* 1st Drive	12600	19800
3570-B* 2nd Drive	13600	20400
3570-C* 1st Drive	21500	42900
3570-C* 2nd Drive	21300	42000
3590* 1st Drive	23500	48800
3590* 2nd Drive	23200	48900
* = Save performed with USEOPTBLK(*YES)		

Table 90. Model S20 feature 2166 Concurrent Restores		
Tape Device	**Work Load**	
	NUMX	2GB
3570-B* 1st Drive	12500	19400
3570-B* 2nd Drive	13200	20000
3570-C* 1st Drive	20900	42400
3570-C* 2nd Drive	21000	41700
3590* 1st Drive	23300	50100
3590* 2nd Drive	23200	48800
* = Save performed with USEOPTBLK(*YES)		

15.10 Save/Restore Rates for Optical Device

The following save and restore performance measurements were made on the 3995 Model C48 Optical Library with a 4X drive. The save rates and the restore rates are expressed in terms of megabytes per hour (MB/HR). They include the time needed to complete the save or restore operation, but not the time that is required for the autochanger to load or unload the optical cartridge.

Operation	**Work Load**	
	NUMX	2GB
Save	1925	2160
Restore	3090	6470

The following save and restore performance measurements were made on the 3995 model C46 Optical Library with an 8X drive.

Operation	**Work Load**	
	NUMX	2GB
Save	3200	3500
Restore	3800	7600

15.11 HIERARCHICAL STORAGE MANAGEMENT

Hierarchical Storage Management (HSM) provides an automatic way of managing and distributing data between the different storage layers to meet the users needs for accessing data while minimizing the overall cost. The concept of HSM involves the placement of data items in such a way as to minimize the cost of storing your data, while maximizing the accessibility.

The following examples were run on a 20S - 2166 System containing 16 - 6607 DASD in each of the ASP′s. One of the ASP′s was Compressed and one was not The rate at which the data was demoted and promoted was approximately the same making HSM the gating factor for performance in this example. The number and type of dasd can and will effect customer performance see the dasd section on this guide for information on dasd types and the effects of compression.

The rates are expressed in megabytes per hour (MB/HR)

Table 91. HSM Rates For Migrating Data Between ASPs

Work Load	**Transfer Rate (MB/HR)**
NUMX	5700
2GB	10000
DLO	2200

15.12 Save/Restore Tips for Better Performance

1. The choice of using hardware data compression (DTACPR), data compaction (COMPACT), and the USEOPTBLK parameter are important in getting the best performance out of your tape drives. The settings in the following table work best on average data. However, you may have to experiment with these settings to get the best performance for your environment.

Table 92. Settings for Optimum Performance

Tape Device	DTACPR	COMPACT	USEOPTBLK
6380	*YES	*NO	*NO
6382/6383	*YES	*NO	*NO
6390	*DEV	*DEV	*NO
7208-342	*DEV	*DEV	*YES
6385[1]	*DEV	*DEV	*NO
3570-B	*DEV	*DEV	*YES
3570-C	*DEV	*DEV	*YES
3590	*DEV	*DEV	*YES

[1] The 6385 can experience performance degradation if the system is unable to keep its buffer filled. In this case you may find better performance with DTACPR(*NO) COMPACT(*NO).

2. For V3R7 and later releases, setting the USEOPTBLK parameter to *YES on save commands can significantly improve performance on newer tape drives. In V4R2 USEOPTBLK(*YES) is the default. This is especially true where the system′s CPU is subjected to a heavy work load. The newer drives are designed to take advantage of the parameter. The drives that make use of this today are displayed in the table in the introduction section of this document.

3. Adjusting the system configuration will be most useful to those using fast tape drives. Keep in mind that following information only applies to optimizing the tape drive for save operations and restore operations. In order to get the fastest saves the tape IOP might not be placed where it can act as an alternate IPL device. In that case you would need to have a slot in the main system unit to move to if you ever needed to restore your system. With Release V4R1 the IPL device is no longer restricted to the main system unit.

   In the following discussion on bus speed, the concern is mainly with the write capability. (Write capability is the rate at which the system can send data to a tape device across that bus). For most of the buses the system can read data from devices much faster than they can write to them. As of V4R3, the 3570-C and 3590 are the only tape drives that can outperform the buses. However, some of the concepts in this section can help other tape drives perform better by avoiding data collisions.

   The placement of the tape and DASD IOP′s are very important to save and restore performance. There are several things to consider when placing the IOP card. The first consideration is the speed of the bus. The copper bus (the bus in the main system unit) will vary in speed, depending on the system model you have. The 5XX models have a 16MB/S write capability while the new S20 has a 25MB/S write capability. The 266 bus from a Model 510 or Model 50S is rated at 14MB/S write capability. The 1063 bus from a Model

530 or larger system is rated at 25MB/S write capability. Note that if you have 5044 expansion units attached to a Model 530, the 266 buses will connect those units.

The tape IOP should be placed in bus 0 on a Model 510 or Model 50S. This placement will provide the best performance from a 3570-B tape device,but will not provide optimum performance from a 3570-C or 3590 tape device. For a Model 530 or Model 53S the IOP should be placed on the 1063 bus that has the least amount of DASD attached to it. This is to avoid collisions with the data that is being gathered and the data that is being sent to the tape drive. On the new S20 systems, the main copper bus or the 1063 may be used, since both are rated at approximately 25MB/S write capability.

If a user wishes to fully drive multiple tape drives, balanced user Auxiliary Storage Pools (ASPs) need to be created. It will also be necessary to know which tape drive is being fed by which ASP. This is to ensure that the tape IOP is placed on the proper bus to avoid the data flowing from the other ASPs. The most likely case will be that the user will have several large data files. Most of the system will be fed to one tape drive while the data files are fed to the other available tape drives. In either case, a tape management system such as BRMS/400 is recommended to keep track of the data and make the most of multiple tape drives. Multiple tape drives can be used without user ASP′s but the performance of the tape drives will be degrade depending upon the distribution of the data and the placement of the tape drives.

4. Bus cables, the optical cables that connect the buses, come in different lengths. The 6 to 20 meter cables do not display much difference in affecting the performance of save operations and restore operations. IBM testing determined that save operations ran 55% slower over a 2 kilometer cable as compared to a 6 meter cable. The 2 kilometer cable is only supported on a 266 bus which is rated at 14MB/S write capability. The 266 bus normally drives the 3590 tape device at around 80% of its capability. When the 2 kilometer cable is added, 3590 tape device is being driven at around 40% of its capability. The 500 meter cable is supported on the 1063 bus which is rated at 25 MB/S write capability. The difference in performance between the 500 meter cable and the 6 meter cable on a 3590 across the 1063 bus is approximately 10% to 15%.

15.13 New For V4R3

- Tape Device 3570 Model C.
- Hierarchical Storage Management (HSM). see section 15.11for details.

Chapter 16. IPL Performance

Performance information for Initial Program Load(IPL) is included in this section. Improvements have been made in V4R3 to reduce the amount of time it takes to IPL an AS/400.

A reduction in Abnormal IPL time (IPL′s where an abrupt shutdown, eg. a power failure, caused the system to do recovery processing in either SLIC, OS/400 or both) of up to 40% may be realized in V4R3 over V4R2 due to improvements in Storage Management Directory Recovery. Depending on the severity of the IPL, the system may not go into full directory recovery, but will only run a sub-set of this.

The primary focus of this section is to present data that compares V4R2 IPL times versus V4R3 IPL times across four different hardware and software configurations. The data for both normal and abnormal IPL′s is broken down into phases, making it easier to see the detail of where the performance gains are realized.

NOTE: The information that follows is based on performance measurements and analysis done in the AS/400 division laboratory. Actual performance may vary significantly from what is provided here.

IPL Performance Considerations

There are many factors that affect IPL performance. The wide variety of hardware configurations and software environments available to an AS/400 customer make it difficult to characterize a ′typical′ IPL environment and predict the results. The following is a high level look at some of the factors that affect IPL duration.

NOTE: This list is by no means exhaustive.

1. Type of IPL performed
   - Normal IPL
     - Power On IPL (cold start)
     - Programmed IPL (PWRDWNSYS RESTART *YES)
   - Abnormal IPL
     - System abnormally terminated causing recovery processing to be done during the IPL. The amount of processing is determined by the point in time the system went down and how it went down (eg. function 3, function 8, function 22, function 34, machine check, ″white button″ power off, power outage with continuous power main storage(CPM), and power outage with no protection).

2. Hardware Configuration
    - CPU - model, feature code
    - DASD - amount, type, number of arms on MFIOP, protection used, size of objects stored, number of ASPs
    - Amount and type of IOPs
    - Amount of main storage
    - Number of devices on system
    - Number of towers
3. Software Configuration
    - Total number of jobs on the system
    - Number of spool files
    - Number of user profiles
    - Number of libraries
    - Use of Journal, Commit, and SMAPP
    - Database size and characteristics
    - Type of active workload

IPL Benchmark Description

For normal IPLs, benchmark time is measured from power-on to console sign-on screen(see note) . Additional IPL times are included for V4R3 when using a programmed IPL(PWRDWNSYS RESTART *YES).

NOTE: Due to some long running tasks (like TCP/IP), all workstations may not be up and ready at the same time that the console workstation is

For abnormal IPLs, the benchmark consists of bringing up a database workload(see configuration charts) and letting it run for a specified period of time. Once the workload is stabilized, a function 22 is issued to the op panel, forcing a main store dump(MSD). This environment simulates a loop or hang situation. The dump is then copied to DASD via the Auto Copy function. The Auto Copy function is enabled thru System Service Tools(SST). Once the dump is copied, the system completes the remaining IPL with no user intervention. This is possible using the Auto Copy function, and by putting the key in normal mode shortly after the function 22 is requested. Benchmark time is measured from the time the function 22 is issued, to the time the console sign on screen appears.

Four hardware and software environments were measured. Common across all four environments was the use of the default settings on the CHGIPLA command which includes having the parameter, HDWDIAG, set to (*MIN). All physical files are explicitly journaled. Also, all logical files are journaled using SMAPP (System Managed Access Path Protection). set to *MIN. This is accomplished by changing the default setting using the EDTRCYAP command.

12-way (2189) System Benchmark Definition

Hardware Configuration
650-2189(12-way) with 40 GB Main Storage
1484 GB DASD / 300 arms, 20 IOPs (20 arms on MFIOP)
DASD in ASP1 was mirrored, all remaining DASD was a combination of RAID protected and unprotected
30 comm IOP′s / 43 workstation IOA′s
200 remote printers
3 ASP′s

Software Configuration
90,000 spool files(30,000 completed jobs with 3 spool files each)
1000 jobs waiting on job queues (active)
9000 delayed jobs waiting on job queues (in-active)
6000 user profiles
3000 libraries
42,600 jobs in system

Database:

- 25 libraries with 2600 physical files and 452 logicals
- 2 libraries with 10,000 physical files and 200 logicals

NOTE:

- All physical files are explicitly journaled
- All logical files are journaled using SMAPP
- SMAPP is set to a minimum
- Commitment Control used on 20% of the files

12-way (2243) System Benchmark Definition

Hardware Configuration
650-2243(12-way) with 20 GB Main Storage
1287 GB DASD / 318 arms (4 GB each), 20 IOPs (20 arms on MFIOP)
MFIOP DASD is mirrored, all other DASD is RAID protected
51 comm IOP′s / 109 workstation IOP′s / (4360 local workstations)
200 remote printers
3 ASP′s

Software Configuration
90,000 spool files(30,000 completed jobs with 3 spool files each)
1000 jobs waiting on job queues (active)
9000 delayed jobs waiting on job queues (in-active)
6000 user profiles
3000 libraries
42,600 jobs in system

Database:

- 25 libraries with 2600 physical files and 452 logicals
- 2 libraries with 10,000 physical files and 200 logicals
- CPW database

NOTE:

- All physical files are explicitly journaled
- All logical files are journaled using SMAPP
- SMAPP is set to a minimum
- Commitment Control used on 20% of the files

8-way System Benchmark Definition

Hardware Configuration
650-2240(8-way) with 16 GB Main Storage
900 GB DASD / 227 arms, 15 IOPs (20 arms on MFIOP)
3200 Local Workstations / 8000 virtual devices
80 workstation IOPs / 18 LAN IOPs
1 FSIOP
200 Remote printers

Software Configuration
90,000 spool files(30,000 completed jobs with 3 spool files each)
10,000 jobs in job queues
6000 user profiles
1000 libraries
42,000 jobs in system

Database:

- 20 libraries with 2000 physical files and 382 logicals
- 2 libraries with 10,000 physical files and 200 logicals
- CPW database
- 3 ASPs

NOTE:

- MFIOP DASD is mirrored, all other DASD units are RAID protected
- Commitment Control used on 20% of the files

Small System Benchmark Definition

Hardware Configuration
170-2291 with 128MB Main Storage
4 arms, 8 GB DASD units, 1 IOP
2 token ring cards
Software Configuration
2,000 spool files(2,000 completed jobs with 1 spool file per job)
350 jobs in job queues
100 user profiles
200 libraries
2,500 jobs in system

Database:

- 1 library with 100 physical files and 20 logicals
- 1 library with 50 physical files and 10 logicals

NOTE:

- All DASD units are unprotected
- Commitment Control is not used on files

IPL Performance Measurements

The following tables provide a comparison summary of the measured performance data for normal and abnormal IPLs. Results presented do not represent any particular customer environment.

Table 93. AS/400 Normal IPL Benchmark Summary - Power-On(Cold Start)

	12-way System 650-2189		**12-way System 650-2243**		**Large (8-way) System 530-2162**		**Small System 170-2291**	
	V4R2	**V4R3**	**V4R2**	**V4R3**	**V4R2**	**V4R3**	**V4R2**	**V4R3**
Hardware	-	10:20	7:54	6:58	5:35	-	-	4:50
SLIC	-	1:49	5:03	2:29	1:55	-	-	0:54
OS/400	-	6:02	10:47	9:48	2:50	-	-	2:07
Total	-	18:11	23:43	19:14	10:25	-	-	7:51

Note:

- Measurement units are in minutes:seconds
- Benchmark is measured from power-on to console sign on
- Generally, the hardware phase is composed of C1xx xxxx and C3xx xxxx SRCs, SLIC is composed of C600 xxxx SRCs, and OS/400 is composed of C900 xxxx SRCs plus time to console sign on.
- For V4R3 normal programmed IPLs(PWRDWNSYS RESTART *YES), the durations would be as follows:
  - 12-way (2189) System - reduces V4R3 cold start IPL time by 0:21
  - 12-way (2243) System - reduces V4R3 cold start IPL time by 1:00
  - Large 8-way System - reduces V4R3 cold start IPL time by 0:15
  - Small System - reduces V4R3 cold start time by 0:03

Table 94. AS/400 Abnormal IPL Benchmark Summary

	12-way System 650-2189		**12-way System 650-2243**		**Large 8-way System 530-2162**		**Small System 170-2291**	
	V4R2	**V4R3**	**V4R2**	**V4R3**	**V4R2**	**V4R3**	**V4R2**	**V4R3**
Processor MSD	-	10:27	2:57	3:04	2:40	-	-	2:01
Hardware IPL	-	1:46	1:11	1:11	1:05	-	-	0:50
SLIC MSD IPL with Copy	-	54:50	48:05	53:07	28:05	-	-	3:02
Directory Recovery Adjust- ment	-	-	1:00:00 (see note below)	-	1:00:00 (see note below)	-	-	-
Shutdown	-	:17	:18	:16	:17	-	-	:18
Hardware re-IPL	-	7:59	4 37	4:23	3:45	-	-	3:12
SLIC re-IPL	-	6:38	8 36	7:24	7:35	-	-	1:36
OS/400	-	23:33	49:19	35:14	33:10	-	-	5:26
Total	-	1:45:19	2:55:02	1:44:39	2:16:45	-	-	16:26

Note:

- Measurement units are in hours:minutes:seconds
- MSD is Main Store Dump
- For the 12-way (2189) system, if RAID is used in ASP 1 instead of mirroring, a degradation in IPL time can be expected.
- The following is a general view of each IPL phase as it relates to the SRCs posted on the operation panel: Processor MSD includes the C1xx xxxx and D1xx xxxx right after the function 22 is issued. Hardware IPL is the next phase which includes the following group of C1xx xxxx and C3xx xxxx SRCs. SLIC MSD IPL with Copy follows with the next series of C6xx xxxx. Directory Recovery is specific to the C600 4260 SRC. The copy occurs during the C6xx 4404 SRCs. Shutdown includes the Dxxx xxxx SRCs. Hardware re-IPL includes the next phase of C1xx xxxx and C3xx xxxx. SLIC re-ipl follows which are the C600 xxxx SRCs. OS/400 completes with the C900 xxxx SRCs plus the time to console sign on.
- Directory Recovery Adjustment - The directory recovery SRC(C600 4260)duration varies widely from one customer environment to another. Duration depends primarily on the following:
  - Number of DASD extents on each DASD
  - Contention between tasks
  - Largest size of individual DASD devices
  - Slowest speed of individual DASD devices
- For V4R2, an artificial adjustment time of one hour was made to the 8-way and 12-way system benchmarks to approximate the duration of an actual customer environment. For V4R3 the time is an actual time.
- ACTUAL CUSTOMER TIMES MAY VARY SIGNIFICANTLY FROM THIS.

Conclusions/Recommendations for IPL

Many AS/400 customers will see a significant improvement in IPL times in V4R3. Although IPL duration is highly dependent on hardware and software configuration, there are tasks that can be performed to reduce the amount of time required for the system to perform an IPL. The following is a partial list of recommendations for optimal IPL performance:

- Reduce the number of jobs on the system. The best way to do this is by removing unnecessary spool files. Use the Display Job Tables (DSPJOBTBL) command to monitor the size of the job table(s) on the system, and the Change IPL Attributes (CHGIPLA) command to compress job tables when there are a large amount of available job table entries.
- Reduce the number of device descriptions - remove any obsolete device descriptions from the system.
- Control the level of hardware diagnostics that are run during an IPL. On the CHGIPLA command, by specifying the default, HDWDIAG(*MIN), the system will perform only a minimum, critical set of hardware diagnostics. This type of IPL is appropriate in most cases. The exceptions include a suspected hardware problem, or when new hardware, such as additional memory, is being introduced to the system.
- Reduce the amount of rebuild time for access paths during an IPL by using System Managed Access Path Protection (SMAPP). The *AS/400 Backup and Recovery* book (SC41-5304) describes this method for protecting access paths from long recovery times during an IPL.
- For additional information on how to improve IPL performance, refer to *AS/400 Basic System Operation, Administration, and Problem Handling* (SC41-5206).

Chapter 17. NT IPCS

This chapter will be updated in October, 1998, when an IBM White Paper becomes available. The ″NT on IPCS″ white paper will be posted on the Competitive/Performance IBM Intranet site at:
http://ca-web.rchland.ibm.com/perform/perfmenu.htm

Chapter 18. Miscellaneous Performance Information

18.1 Public Benchmarks (TPC-C, SAP, RPMark, NotesBench)

TPC-C Commercial Performance

The Transaction Processing Performance Council′s TPC Benchmark C (TPC-C (**)) is a public benchmark that stresses systems in a full integrity transaction processing environment. It is more closely related to general business computing than prior benchmarks (such as TPC-A and TPC-B), but the functional emphasis may still vary significantly from an actual customer environment. For additional information on the benchmark and current results, please refer to the TPC′s web site at http://www.tpc.org.

In addition to referring to the TPC web site, request the ′AS/400 TPC-C Results′ package from MKTTOOLS for a set of questions and answers about the AS/400 TPC-C results. To request this package issue the following command on VM:

```
TOOLCAT MKTTOOLS GET AS4TPCC PACKAGE
```

Information for can be found at:
http://www.tpc.org/bench.results.html

SAP Performance Information

In September, 1995, SAP AG of Walldorf, Germany announced availability of its R/3 suite of client/server business applications to IBM′s AS/400 Advanced Series. The pilot phase of R/3 release 3.0 for AS/400 began in fourth quarter ′95, with a controlled availability phase beginning in 3/96 and general availability in 7/96. SAP′s R/3 suite of client/server applications includes solutions for manufacturing, sales and distribution, financial accounting and human resource processes.

R/3 release 3.0 for AS/400 will provide a scalable solution for those customers wishing to manage their business processes on the AS/400.

Two benchmarks are primarily used today for making platform comparisons and sizing estimates of the R/3 product. The first is called FI and exercises the financial portion of the product. The second is called SD and exercises the sales and distribution portion of the product. Sizing information based on these benchmarks for R/3 release 3.0 is available from the IBM SAP Competency Centers.

For further details concerning a particular SAP configuration or for additional questions, please contact an IBM SAP Competency Center. The North America Competency Center can be reached at WASVMIC1(IBMSAPCC) or 1-800-426-0222. The International Competency Center can be reached at MUNIVM4(ISICC) or 49-6227-34-1298.

A paper describing current 2-tier and 3-tier benchmark results can be found at:
http://www.sap.com/products/techno/media/pdf/wp_be2_e.pdf

RPMark95

Information can be found at:
http://www.cslinc.com/rpmark/rpmark.htm

NotesBench

Information can be found at:
http://www.notesbench.org

18.2 Dynamic Priority Scheduling

On an AS/400 CISC-model, all ready-to-run OS/400 jobs and Licensed Internal Code (LIC) tasks are sequenced on the Task Dispatching Queue (TDQ) based on priority assigned at creation time. In addition, for N-way models, there is a cache affinity field used by Horizontal Licensed Internal Code (HLIC) to keep track of the processor on which the job was most recently active. A job is assigned to the processor for which it has cache affinity, unless that would result in a processor remaining idle or an excessive number of higher-priority jobs being skipped. The priority of jobs varies very little such that the resequencing for execution only affects jobs of the same initially assigned priority. This is referred to as Fixed Priority Scheduling.

For V3R6, the new algorithm being used is Dynamic Priority Scheduling. This new scheduler schedules jobs according to ″delay costs″ dynamically computed based on their time waiting in the TDQ as well as priority. The job priority may be adjusted if it exceeded its resource usage limit. The cache affinity field is no longer used in a N-way multiprocessor machine. Thus, on an N-way multiprocessor machine, a job will have equal affinity for all processors, based only on delay cost.

A new system value, QDYNPTYSCD, has been implemented to select the type of job dispatching. The job scheduler uses this system value to determine the algorithm for scheduling jobs running on the system. The default for this system value is to use Dynamic Priority Scheduling (set to ′1′). This scheduling scheme allows the CPU resource to be spread to all jobs in the system. The benefits of Dynamic Priority Scheduling are:

- No job or set of jobs will monopolize the CPU
- Low priority jobs, like batch, will have a chance to progress
- Jobs which use too much resource will be penalized by having their priority reduced
- Jobs response time/throughput will still behave much like fixed priority scheduling

By providing this type of scheduling, long running, batch-type interactive transactions, such as a query, will not run at priority 20 all the time. In addition, batch jobs will get some CPU resources rather than interactive jobs running at high CPU utilization and delivering response times that may be faster than required.

To use Fixed Priority Scheduling, the system value has to be set to ′0′.

Delay Cost Terminology

- Delay Cost

  Delay cost refers to how expensive it is to keep a job in the system. The longer a job spends in the system waiting for resources, the larger its delay cost. The higher the delay cost, the higher the priority. Just like the priority value, jobs of higher delay cost will be dispatched ahead of other jobs of relatively lower delay cost.

- Waiting Time

  The waiting time is used to determine the delay cost of a job at a particular time. The waiting time of a job which affects the cost is the time the job has been waiting on the TDQ for execution.

- Delay Cost Curves

  The end-user interface for setting job priorities has not changed. However, internally the priority of a job is mapped to a set of delay cost curves (see “Priority Mapping to Delay Cost Curves” below). The delay cost curve is used to determine a job′s delay cost based on how long it has been waiting on the TDQ. This delay cost is then used to dynamically adjust the job′s priority, and as a result, possibly the position of the job in the TDQ.

  On a lightly loaded system, the jobs′ cost will basically stay at their initial point. The jobs will not climb the curve. As the workload is increased, the jobs will start to climb their curves, but will have little, if any, effect on dispatching. When the workload gets around 80-90% CPU utilization, some of the jobs on lower slope curves (lower priority), begin to overtake jobs on higher slope curves which have only been on the dispatcher for a short time. This is when the Dynamic Priority Scheduler begins to benefit as it prevents starvation of the lower priority jobs. When the CPU utilization is at a point of saturation, the lower priority jobs are climbing quite a way up the curve and interacting with other curves all the time. This is when the Dynamic Priority Scheduler works the best.

  Note that when a job begins to execute, its cost is constant at the value it had when it began executing. This allows other jobs on the same curve to eventually catch-up and get a slice of the CPU. Once the job has executed, it ″slides″ down the curve it is on, to the start of the curve.

Priority Mapping to Delay Cost Curves

The mapping scheme divides the 99 ′user′ job priorities into 2 categories:

- User priorities 0-9

  This range of priorities is meant for critical jobs like system jobs. Jobs in this range will NOT be overtaken by user jobs of lower priorities. NOTE: You should generally not assign long-running, resource intensive jobs within this range of priorities.

- User priorities 10-99

  This range of priorities is meant for jobs that will execute in the system with dynamic priorities. In other words, the dispatching priorities of jobs in this range will change depending on waiting time in the TDQ if the QDYNPTYSCD system value is set to ′1′.

The priorities in this range are divided into groups:

- Priority 10-16
- Priority 17-22
- Priority 23-35
- Priority 36-46
- Priority 47-51
- Priority 52-89
- Priority 90-99

Jobs in the same group will have the same resource(CPU seconds and Disk I/O requests) usage limits. Internally, each group will be associated with one set of delay cost curves. This would give some preferential treatment to jobs of higher user priorities at low system utilization.

With this mapping scheme, and using the default priorities of 20 for interactive jobs and 50 for batch jobs, users will generally see that the relative performance for interactive jobs will be better than that of batch jobs, without CPU starvation.

Performance Testing Results

Following are the detailed results of two specific measurements to show the effects of the Dynamic Priority Scheduler:

In Table 95, the environment consists of the RAMP-C interactive workload running at approximately 70% CPU utilization with 120 workstations and a CPU intensive interactive job running at priority 20.

In Table 96 on page 297, the environment consists of the RAMP-C interactive workload running at approximately 70% CPU utilization with 120 workstations and a CPU intensive batch job running at priority 50.

Table 95. Effect of Dynamic Priority Scheduling: Interactive Only

	QDYNPTYSCD = ′1′ (ON)	**QDYNPTYSCD = ′0′ (OFF)**
Total CPU Utilization	93.9%	97.8%
Interactive CPU Utilization	77.6%	82.2%
RAMP-C Transactions Per Hour	60845	56951
RAMP-C Average Response Time	0.32	0.75
Priority 20 CPU Intensive Job CPU	21.9%	28.9%

Table 96. Effect of Dynamic Priority Scheduling: Interactive and Batch

	QDYNPTYSCD = ′1′ (ON)	**QDYNPTYSCD = ′0′ (OFF)**
Total CPU Utilization	89.7%	90.0%
Interactive CPU Utilization	56.3%	57.2%
RAMP-C Transactions Per Hour	61083	61692
RAMP-C Average Response Time	0.30	0.21
Batch Priority 50 Job CPU	15.0%	14.5%
Batch Priority 50 Job Run Time	01:06:52	01:07:40

Conclusions/Recommendations

- When you have many interactive jobs running on the system and want to ensure that no one CPU intensive interactive job ′takes over′ (see Table 95 on page 296), Dynamic Priority Scheduling will give you the desired result. In this case, the RAMP-C jobs have higher transaction rates and faster response times, and the priority 20 CPU intensive job consumes less CPU.
- Dynamic Priority Scheduling will ensure your batch jobs get some of the CPU resources without significantly impacting your interactive jobs (see Table 96). In this case, the RAMP-C workload gets less CPU utilization resulting in slightly lower transaction rates and slightly longer response times. However, the batch job gets more CPU utilization and consequently shorter run time.
- It is recommended that you run with Dynamic Priority Scheduling for optimum distribution of resources and overall system performance.

For additional information, refer to the *Work Management Guide*

18.3 Main Storage Sizing Guidelines

To take full advantage of the performance of the new AS/400 Advanced Series using PowerPC technology, larger amounts of main storage are required. To account for this, the new models are provided with substantially more main storage included in their base configurations. In addition, since more memory is required when moving to RISC, memory prices have been reduced.

The increase in main storage requirements is basically due to two reasons:

- When moving to the PowerPC RISC architecture, the number of instructions to execute the same program as on CISC has increased. This does not mean the function takes longer to execute, but it does result in the function requiring more main storage. This obviously has more of an impact on smaller systems where fewer users are sharing the program.
- The main storage page size has increased from 512 bytes to 4096 bytes (4KB). The 4KB page size is needed to improve the efficiency of main storage management algorithms as main storage sizes increase dramatically. For example, 4GB of main storage will be available on AS/400 Advanced System model 530.

The impact of the 4KB page size on main storage utilization varies by workload. The impact of the 4KB page size is dependent on the the way data is processed. If data is being processed sequentially, the 4KB page size will have little impact on main storage utilization. However, if you are processing data randomly, the 4KB page size will most likely increase the main storage utilization.

The minimum memory available on RISC systems is 32MB versus 8MB on CISC systems. In most instances, 8MB CISC systems will require 32MB on RISC when running the same workload. However, if the 8MB CISC system is overcommitted in main storage utilization, then 64MB of main storage may be required on RISC.

As a first approximation of the main storage required when moving to AS/400 models using PowerPC AS technology, use the following guidelines. If you will be adding additional work as you upgrade to the new models, you should first determine what main storage would be required on CISC for this new workload before using the guidelines below.

Table 97. Main Storage Size Guidelines

Main Storage Size on CISC	Main Storage Size on RISC
Up to 160MB	(2 X CISC Main Storage) + 16MB (See note below)
Greater than 160MB	2 X CISC Main Storage

Note:

The 16MB that is added is primarily due to the increase in size of the operating system code that must be resident in main storage. It is very important to take this increase into account when sizing systems with lower amounts of main storage.

This initial estimate should be followed up by a more detailed analysis with BEST/1 for OS/400. Using BEST/1 to analyze your workload will take into account how main storage is being utilized on your current system.

Refer to 13.4, "Working Memory Guidelines for Compiles" on page 231 for main storage sizing considerations for application development environments.

18.4 Memory Tuning Updates

The Performance Adjustment support (QPFRADJ system value) that is used for initially sizing memory pools and managing them dynamically at run time has been enhanced to support the new RISC hardware and V3R6. In addition, at V3R7 the CHGSHRPOOL and WRKSHRPOOL commands have been updated so that you can tailor memory tuning parameters used by QPFRADJ. Now you can specify your own faulting guidelines, storage pool priorities, and minimum/maximum size guidelines for each pool. This allows you the flexibility to set unique QPFRADJ parameters at the pool level.

The following changes were made for tuning done at IPL time (the system value QPFRADJ is set to 1 or 2):

1. The calculation for the minimum Machine pool size has been updated to reflect changes in the amount of storage needed for lines, controllers, and devices. The algorithm also has been changed to use (as a base value) the minimum Machine pool size calculated by the License Internal Code (LIC)

instead of the tabular method (based on main storage size) that was used in previous releases.

2. The pool size calculation for the *INTERACT and *BASE pools has been updated. After the Machine and Spool pool sizes have been determined, 70% of the remaining storage is given to *INTERACT. The remaining 30% is given to the *BASE pool. To better support the unique demands of the client/server environment, on Server models the amounts are the opposite (70% to *BASE, 30% to *INTERACT).

The following changes were made to dynamic tuning done at run time - (the system value QPFRADJ is set to 2 or 3):

With V3R6, the page fault guidelines stated in the Work Management Guide (SC41-4306) are significantly higher than they are for V3R1. The large increases are due to now including index faults in the count for V3R6, such as faults on storage management directories, user profiles, libraries, and file access paths.

1. The Dynamic Tuner no longer uses the guidelines published in the Work Management Guide for User pools (any pool except the Machine pool). The guidelines are on the average good, but the tuner is able to take more into account at run time to adjust the faulting guidelines based on workload characteristics. The Dynamic Tuner now calculates a run-time guideline based on the number of active jobs in the storage pool. The type of pool (*INTERACT, *SPOOL, or other shared pool) is also taken into consideration. The run-time guidelines could be much different than the published guidelines, especially if the number of jobs is much lower or much higher than the number of jobs for which the system is rated.
2. For Advanced Server models, the *BASE pool will be treated with higher priority than the *INTERACT pool. This means that if the Dynamic Tuner determines that both *BASE and INTERACT require more memory, *BASE will get more memory before *INTERACT. On traditional models, *INTERACT gets higher priority.
3. The minimum pool size for an active pool has been increased.
   - *INTERACT - 3000K
   - *SPOOL - 256K
   - *SHRPOOL1-10 - 1000K
   - If inactive, pools may be temporarily reduced to 256K (except the Machine and Base pool).
4. The maximum pool size for an active pool has been increased to 3072K for each active job in the pool (up from 2048K).

18.5 User Pool Faulting Guidelines

Due to the large range of AS/400 processors and due to an ever increasing variance in the complexity of user applications, paging guidelines for user pools are no longer published. Only machine pool guidelines and system wide guidelines (sum of faults in all the pools) are published. Even the system wide guidelines are just that...guidelines. Each customer needs to track response time, throughput, and cpu utilization against the paging rates to determine a reasonable paging rate.

There are two choices for tuning user pools:

1. Set system value QPFRADJ = 2 or 3. This algorithm has been changed for V3R6 and PTFed back to V3R1. The new algorithm is much better in several ways, including how it determines which pool has a paging problem, and the speed in which it can react to the needs of those pools with problems. Many customers are reporting they now don′t have to worry about pool tuning at all. They still may need to analyze whether they need to increase total main storage. The rest of this section will help with that analysis.
2. Manual tuning. Move storage around until the response times and throughputs are acceptable. The rest of this section deals with how to determine these acceptable levels.

To determine a reasonable level of page faulting in user pools, determine how much the paging is affecting the interactive response time or batch throughput. These calculations will show the percentage of time spent doing page faults.

The following steps can be used: (all data can be gathered w/STRPFRMON and printed w/PRTSYSRPT). The following assumes interactive jobs are running in their own pool, and batch jobs are running in their own pool.

Interactive:

1. flts = sum of database and non-database faults per second during a meaningful sample interval for the interactive pool.
2. rt = interactive response time for that interval.
3. diskRt = average disk response time for that interval.
4. tp = interactive throughput for that interval in transactions per second. (transactions per hour/3600 seconds per hour)
5. fltRtTran = diskRt * flts / tp = average page faulting time per transaction.
6. flt% = fltRtTran / rt * 100 = percentage of response time due to
7. If flt% is less than 10% of the total response time, then there′s not much potential benefit of adding storage to this interactive pool. But if flt% is 25% or more of the total response time, then adding storage to the interactive pool may be beneficial (see NOTE below).

Batch

1. flts = sum of database and non-database faults per second during a meaningful sample interval for the batch pool.
2. flt% = flts * diskRt X 100 = percentage of time spent page faulting in the batch pool. If multiple batch jobs are running concurrently, you will need to divide flt% by the number of concurrently running batch jobs.
3. batchcpu% = batch cpu utilization for the sample interval. If higher priority jobs (other than the batch jobs in the pool you are analyzing) are consuming a high percentage of the processor time, then flt% will always be low. This means adding storage won′t help much, but only because most of the batch time is spent waiting for the processor. To eliminate this factor, divide flt% by the sum of flt% and batchcpu%. That is:
   newflt% = flt% / (flt% + batchcpu%)
   This is the percentage of time the job is spent page faulting compared to the time it spends at the processor.

4. Again, the potential gain of adding storage to the pool needs to be evaluated. If flt% is less than 10%, then the potential gain is low. If flt% is greater than 25% then the potential gain is high enough to warrant moving main storage into this batch pool.

NOTE:

It is very difficult to predict the improvement of adding storage to a pool, even if the potential gain calculated above is high. There may be instances where adding storage may not improve anything because of the application design. For these circumstances, changes to the application design may be necessary.

Also, these calculations are of limited value for pools that have expert cache turned on. Expert cache can reduce I/Os given more main storage, but those I/Os may or may not be page faults.

18.6 Cryptography Performance

This section provides performance information for AS/400 Common Cryptographic Architecture Services/400 Version 3 (*). This information can be used to assist capacity planning for an AS/400. This data is not representative of a specific customer environment. Results in other environments may vary significantly.

This evaluation was completed on CISC models, but the relative performance and recommendations are similar.

Workload Description

CL and ILE/C application programs were used to utilize the cryptographic functions.

- Encipher and decipher a string of characters 100 times (DEA: Data Encryption Algorithm) with the key specified as:
  - key TOKEN
  - key label located in KEYSTORE
- CPB (“CPB and NetPerf Benchmark Descriptions” on page 79)
  - multiple send/receives pairs of 100 bytes
  - large data transfers

For information on CCAS/400 function, refer to the *Common Cryptographic Architecture Services/400 Installation and Operator's Guide*, SC41-0102.

Measurement Technique

The performance measurements were taken on a dedicated AS/400 (i.e., no other system load from other users) with an application program executing the primitive workload scenarios, previously described. For the communications environment, a second AS/400 was similarly configured. CPU utilization and run time were collected using the WRKSYSSTS command. CPU time is calculated (run time * CPU utilization = CPU time).

Measurement Results

Table 98. Cryptographic Performance

AS/400 Cryptographic Performance
Encipher and Decipher (100 times) from an application
AS/400 D50, V3R1, 2620 Cryptographic Processor

Data Size (Bytes)	Key Type	Run Time (sec)	CPU Util (%)	Delay (ms per pair)	CPU ms per pair	Operations per second
64	TOKEN	7	28.1	70	20	29
1024	TOKEN	7	26.2	70	19	29
2048	TOKEN	8	23.9	80	19	25
4096	TOKEN	10	20.7	100	21	20
8192	TOKEN	13	16.1	130	21	15
16384	TOKEN	27	13.5	270	37	7
64	KEYSTORE	22	78.4	220	173	9
1024	KEYSTORE	28	79.6	280	223	7
2048	KEYSTORE	28	77.1	280	216	7
4096	KEYSTORE	30	73.3	300	220	7
8196	KEYSTORE	34	65.8	340	224	6
16384	KEYSTORE	48	49.8	480	239	4

Note: 1 pair is 2 operations (1 encipher and 1 decipher)

Table 99. SNA Session Level Encryption

SNA Session Level Encryption (SLE)
Application Program Transfer
CPB Benchmark with APPC and the ICF Interface
AS/400 F25, V3R2, 2620 IOP, 2619 TRLAN IOP

Scenario	Time (sec)	Data Rate (kbps)	CPU Time (sec)	CPU Util (%)
10,000 100-byte snd/rcv				
SLE *ALL	1559	10	160	10
SLE *NONE	245	65	94	38
32M large transfer (send)				
SLE *ALL	259	1036	25	10
SLE *NONE	28	9585	11	40

Note:
(1) Receives perform similarly to sends

Conclusions/Explanations

1. The CPU issues requests to the cryptographic processor (CP). The CP then executes the cryptographic function and returns the results to the CPU. Because the CP handles these processing-intensive functions, the CPU is available to process other system activity.

2. The delay time and CPU time are roughly equivalent for both encipher and decipher functions. Therefore, the times listed for encipher and decipher pairs can be cut in half to estimate a single function (first table).

3. As the size of the encipher/decipher request (in bytes) increases, the delay time and CPU time increase (first table).

4. A key is presented to the CCAS/400 API as a 64-byte internal key token or as a key label of an internal key token which is located in a physical file, called key store.

   - Encipher/decipher with a key specified as a TOKEN is faster and consumes less CPU time than when the key is specified as a label.

   - Encipher/decipher with a key specified as a label located in KEYSTORE takes significantly longer and uses significantly more CPU time.

5. MAC (message authentication code) generate and MAC verify yield similar performance results to encipher and decipher. From lab measurements not provided here, trends from the different key specifications, absolute delay times, and CPU times are similar.

6. Note that the performance for the 16384-byte requests seem to have a higher than expected CPU time. This is because the maximum size that can be requested from the CPU to CP is 12K bytes. Therefore, requests that are larger than 12K bytes are broken up by the CCAS/400 API.

7. The CP can handle multiple cryptographic requests from different applications concurrently. From the ′operations per second′ column in the table, note that the rates listed are for that single job (up to 29 operations per second. If multiple instances of the application were run, the aggregate rate would be higher. Being the performance tools do not provide the 2620 Cryptographic Processor utilization, it is difficult to project the capacity of the CP for particular customer scenarios. For small-sized operations, the IOP is limited to about 40 operations per second. For larger-sized operations, the IOP is limited to about 130K bytes per second.

8. When multiple cryptographic requests are sent to the CP from different jobs, they are processed serially in a first come, first served order. Note that if a long request arrives at the CP, it may delay requests from other jobs. This is particularly true with some of the PKA (Public-Key Algorithm) functions. If a job is performance sensitive, it can enter exclusive mode to prevent any other jobs from issuing cryptographic requests.

9. The 2628 Cryptographic Processor has similar performance characteristics to the 2620 CP. When processing requests, the 2628 CP uses the Commercial Data Masking Facility (CDMF) rather than the data encryption algorithm (DEA) that is used by the 2620 CP.

10. **SNA Session Level Encryption (SLE)** provides cryptography functions transparent to the application. By configuring a session for SLE (SLE parameter on the mode description), all transmissions for that session will be enciphered/deciphered. By using a CP, the CPU off-loads this CPU-intensive processing. There is, however, still a significant performance overhead for SLE (second table):

    - **Run Time**: Each time the application sends (or receives) data it must pass the data through the CP. Without SLE, a 100-byte send/receive pair takes 24.5 ms; with SLE, it takes 155.9 ms (6 times longer). Being each send/receive pair transaction uses the CP four times (twice on the system under test, twice on the other AS/400), each use of the CP adds about 33 ms of additional delay. This seems like a large percentage increase here, as this application does no processing other than sending data. For real applications that are more complex, this extra delay may be less significant.

The delay times will vary based on the size and type of the request to the CP and the CPW (relative system performance metric) of the CPU.

- **CPU Time**: When SLE is used, there is also additional processing on the CPU to facilitate getting the data to and from the CP. For the send/receive scenario, SLE used 1.7 times more CPU. For the large transfer scenario, SLE used 2.3 times more CPU. Again, depending on the application that is associated with the transmissions, this overhead may be significant or unnoticeable.
- **Data transfer rate**: For large transfers, the impact of using SLE is significant. For the case in the second table, SLE slowed the data rate by almost 10-fold. For LAN environments (very efficient, and high speed), the effect of SLE is significant. For WAN environments with linespeeds of 64 kbps or slower, the slow-down effect of SLE may be insignificant. Actual data rates will vary based on the size and type of the transmission.

11. There are trade-offs to be considered with encryption over communications. For best performance, use a limited amount of encryption from the application (with the CP) and then send the data. For maximum encryption and ease-of-use, use SLE. If most of the data is to be encrypted, use SLE; SLE provides better performance than application (with the CP) encryption in this case. For a send/receive pair, note from the first table that application encryption uses 20 ms of CPU time. Note from the second table that SLE consumes only 16 ms of CPU time per send/receive pair.

18.7 AS/400 NetFinity Capacity Planning

Performance information for AS/400 NetFinity attached to a V4R1 AS/400 is included below. The following NetFinity functions are included:

- Time to collect software inventory from client PCs
- Time to collect hardware inventoy from client PCs

The figures below illustrate the time it takes to collect software and hardware inventory from various numbers of client PCs. This test was conducted using the Rochester development site, during normal working hours with normal activity (ie. not a dedicated environment). This environment consists of:

- 16 and 4Mb token ring LANs (mostly 16)
- LANs connected via routers and gateways
- Dedicated AS/400
- TCP/IP
- Client PCs varied from 386s to Pentiums (mostly 100 MHz with 32MB memory), using OS/2, Windows/95 and NT
- About 20K of data was collected, hardware and software, for each client

While these tests were conducted in a typical work envirnoment, results from other environments may vary sigificantly from what is provided here.

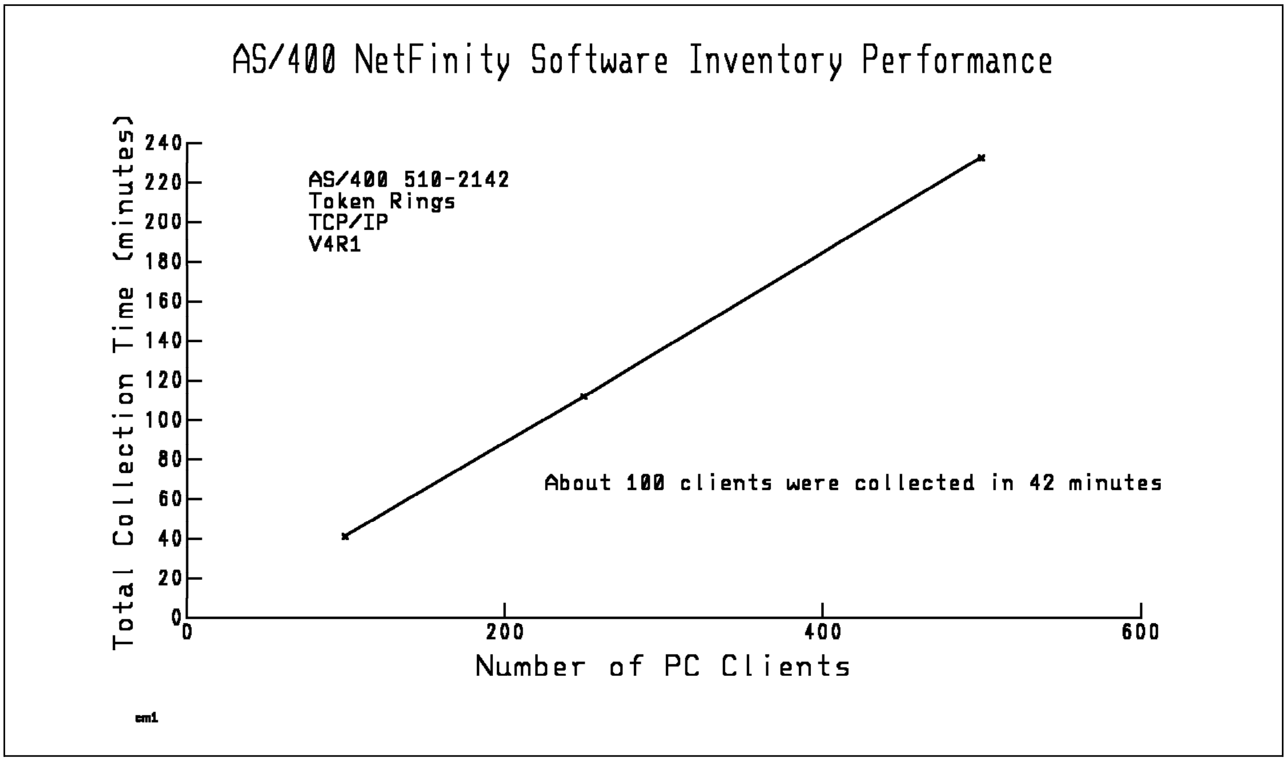


Figure 58. AS/400 NetFinity Software Inventory Performance

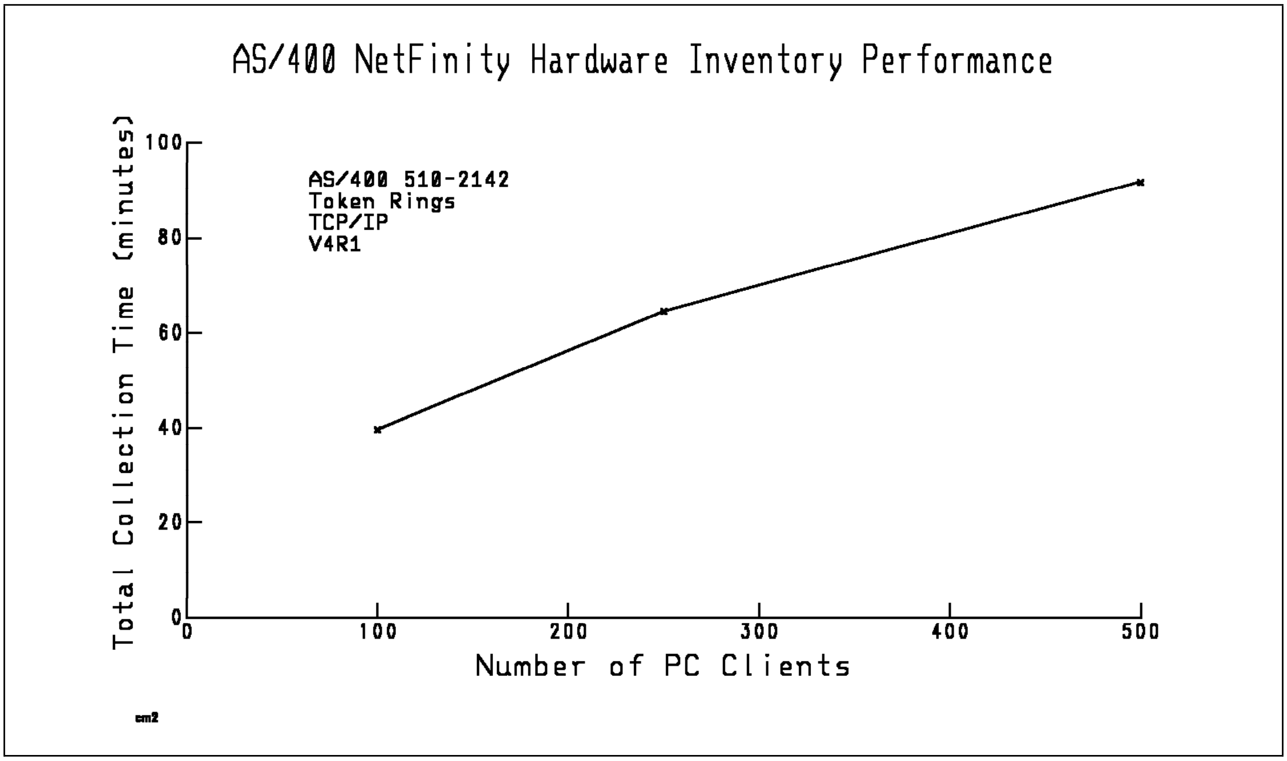


Figure 59. AS/400 NetFinity Hardware Inventory Performance

Conclusions/Recommendations for NetFinity

1. The time to collect hardware or software information for a number of clients is fairly linear.
2. The size of the AS/400 CPU is not a limitation. Data collection is performed at a batch priority. CPU utilization can spike quite high (ex. 80%) when data is arriving, but in general is quite low (ex. 10%).
3. The LAN type (4 or 16Mb Token Ring or Ethernet) is not a limitation. Hardware collection tends to be more chatty on the LAN than software collection, depending on the hardware features.
4. The communications protocol (IPX, TCP/IP, or SNA) is not a limitation.
5. Collected data is automatically stored in a standard DB/2/400 database file, accessible by SQL and other APIs.
6. Collection time depends on clients being powered-on and the needed software turned on. The server will retry 5 times.
7. The number of jobs on the server increases during collection and decreases when not needed.

Appendix A. CPW Benchmark Description

CPW is designed to evaluate a computer system and associated software in the commercial environment. It is rigidly defined for function, performance metrics, and price/performance metrics. It is NOT representative of any specific environment, but it is generally applicable to the commercial computing environment.

- What CPW is
  - Test of a range of data base applications, including simple and medium complexity updates, simple and medium complexity inquiries, realistic user interfaces, and a combination of interactive and batch activities.
  - Test of commitment control
  - Test of concurrent data access by large numbers of users running a single group of programs.
  - Reasonable approximation of a steady-state, data base oriented commercial application.
- What CPW is not:
  - An indication of the performance capabilities of a system for any specific customer situation
  - A test of ″ad-hoc″ (query) data base performance
- When to use CPW data
  - Approximate product positioning between different AS/400 models where the primary application is expected to be oriented to traditional commercial business uses (order entry, payroll, billing, etc.) using commitment control

CPW Application Description

There are five business functions of varying complexity that are simulated in the CPW application. Each simulated user has access to all five functions and must exercise all five with a weighted-random sequence during the course of a measurement. A ″business transaction″ is defined as a combination of a menu selection transaction and one of the five transactions described below.

1. **New Order Transaction**

   The New-Order transaction enters a customer order for 5 to 15 items from a set of supply warehouses. It is a medium-weight read-write transaction that is the foundation of the CPW benchmark. The transaction is executed approximately 43% of the time from each terminal. There is a restriction that 90% of the transactions complete within a 5-second end-user response time.

   The transaction accesses most of the files in the data base with a combination of reads, read-with-updates, and insertion of new records

   Some of the transactions are required to fail, measuring the effect of transaction recovery on the overall system performance.

   The New-Order transaction uses a display screen that uses a majority of a 24x80 character display.

2. **The Payment Transaction**

   Like the New-Order transaction, the Payment transaction is executed approximately 43% of the time.

The basic transaction accesses 3 files by retrieving and updating a single record in each file and inserts a record into a fourth file. 60% of the time, additional reads of a file are required to determine the appropriate key values for the accesses of the basic transaction. This simulates the selection of a customer name from a list of names that match input from an end-user.

The Payment transaction also uses a full-screen interface, although it is not as complex as the New-Order transaction′s screen. Most of the Payment display is used for presenting output of information that is read from the files.

3. **The Order Status Transaction**

   The Order-Status transaction is a medium-weight read-only transaction that has a relatively low frequency of execution (4%).

   The input on the display is quite simple, with the bulk of the transaction spent reading from 3 files and formatting a full screen of output information. Like the payment transaction, some of the time the input requires arbitrary selection of a ″customer″ from a list (requiring input of multiple customers with duplicate last names) and some of the time the input will specify an exact set of key-values to be used to retrieve the order-status information.

4. **The Delivery Transaction**

   The Delivery transaction has two parts: a trivial interactive request that identifies a warehouse and carrier to be used to make deliveries and a more significant batch operation that identifies ″deliverable″ orders assigned to that warehouse and processes a set of 10 of those orders. It is a medium-weight read-write transaction that has a relatively low frequency of execution (4%). The transaction is intended to represent a longer-running (under 80 seconds) operation that would not normally be done at an interactive display, but would often be done at the same time as higher-priority interactive tasks are done and would operate against the same data base as the interactive transactions.

   The Delivery transaction accesses six of the data base files with a combination of read, read-update, and delete. It also posts information in a log file (insert), primarily for the purpose of maintaining statistics for validation of the benchmark.

5. **The Stock Level Transaction**

   The Stock-level transaction is a heavy-weight read-only transaction that has a low frequency of execution (4%). It has very simple input and output to the display screen, but it includes up to 400 read operations across 3 files and it generates a count of entries that satisfy some specific criteria. While not as complex as a true inventory-control-type application, it exercises the same type of file I/O and logic that would be associated with such an application.

CPW Data Base

There are nine data base files used in the CPW benchmark. The files are widely varied in size and are required to scale with the amount of throughput that is claimed for a given system. Some of the files in CPW experience updates to their indexes as well as to the content of their data. This is done through the use of inserts and deletions to the files during the course of the benchmark. All fields in the files contain information that is used by some portion of the benchmark.

1. **Item File**
   - 50,000 records - constant regardless of size of rest of data base
   - 1 key field
   - 4 fields total
   - Records of 35-60 bytes
   - Read by New-order
2. **Warehouse File**
   - 1 record per warehouse (10 terminals are required to support each warehouse)
   - 1 key field
   - 9 fields total
   - Records of 50-80 bytes
   - Read by New-order, updated by Payment
3. **District File**
   - 10 records per warehouse (1/terminal)
   - 2 key fields
   - 11 fields total
   - Records of 70-100 bytes
   - Read by Stock-level, updated by New-order and Payment
4. **Customer File**
   - 30,000 records per warehouse
   - 3 key fields
   - 21 fields total
   - Records of 240-370 bytes
   - Read by New-order, Payment, and Order-status, and updated by Payment and Delivery
5. **Stock File**
   - 50,000 records per warehouse
   - 2 key fields
   - 17 fields total
   - Records of 300-320 bytes
   - Read by Stock-level, updated by New-order
6. **History File**
   - Starts at 30,000 records per warehouse
   - Each Payment transaction adds one record
   - 3 key fields
   - 6 fields total
   - Records of 35-50 bytes
   - Added to by Payment
7. **Order File**
   - Starts with 30,000 records per warehouse
   - Each New-order transaction adds one record
   - 4 key fields
   - 8 fields total
   - Records of about 30 bytes
   - Added to by New-order, read by Order-status, updated by Delivery
8. **Order-Line File**
   - Starts with 300,000 records per warehouse

- Each New-order transaction adds an average of 10 records
- 4 key fields
- 10 fields total
- Records of about 50 bytes
- Added to by New-order, read by Order-status and Stock-level, updated by Delivery

9. **New-Order File**

- Starts with 2,000 records per warehouse
- Continuously added to and deleted from
- 3 key fields
- 3 total fields
- Records of about 10 bytes
- Added to by New-order, deleted from by Delivery

CPW Benchmark Summary

This table summarizes activities performed within the CPW benchmark:

Transaction Type	Complexity	Files Accessed	Logi-cal	Activity
New Order	Complex	Warehouse Customer District Order New-Order Item Stock Order-Line	1 1 2 1 1 10 20 10	Get warehouse tax Get customer discount Get & update next order ID Add a record Add a record Find cost of 10 items Adjust stock qty and YTD ordered Add 10 records
Payment	Simple	Warehouse District Customer History	2 2 2 1	Get & update warehouse YTD Get & update district YTD Get & update YTD balance & count Add one record
Order Status	Medium	Customer Order Order-Line	1 1 10	Get balance Get order list Get all order lines
Delivery	Complex	New-Order Order Order-Line Customer	20 20 200 20	Get & delete 10 records Get/update 10 rcds w/carrierID Get & update 100 rcds w/date Get/update balance&delvry count
Stock-level	Medium to Complex	District Order-line Stock	1 200 200	Get next order ID get lines from last 20 orders get qty on hand from 200 items

CPW Required Functions

- **ACID Properties**

  Atomicity, Consistency, Isolation, and Durability properties are defined as the minimal integrity required of any system for which benchmark results are to be published. Essentially, these properties are designed to ensure that the system under test can guarantee that transaction integrity will be maintained (either all or none of a transaction is committed to the data base) through normal processing and in the event of a single point of failure (such as a power failure or disk failure).

- **Configuration Size Requirements**

  For each warehouse configured in the data base, all the other files (except the constant Item file) must be scaled according to the descriptions listed above. Also, 10 workstations must be configured on the system and must exercise the applications during the benchmark period.

  For the claimed throughput rating of the system measured, sufficient disk space must be configured to hold 180 days of historical data (from the continuously expanding History, Order, and Order-line files) and its associated indexes. There must also be sufficient space in the journal log files to run at the claimed throughput for an 8-hour period.

- **Other Requirements**

  The benchmark specification allows flexibility in implementation, as long as the functions described in the transaction descriptions are completed. Each transaction has a specific set of actions that must be taken using data from the input screen and from the data base files. The means of achieving those actions is left largely to the implementer.

  The design of each workstation screen is more rigidly specified than the application logic. Each screen is described in detail within the specification. The screen interface is required to support such functions as moving the cursor from field to field, correcting data that has already been keyed, requiring input in some fields, and other things that are associated with a full-screen interface.

  The keyed entry for each transaction is also rigidly specified. For any given transaction, specific rules are defined for the rate of keying input, the type and value (often a random number or weighted-random number) of the input, and the amount to delay after a response is received before selecting the next transaction.

  Transaction selection is also rigidly specified. At least 43% of the transactions must be Payment transactions and at least 4% each of the transactions must be Order-status, Delivery, and Stock-level transactions. The remaining 45% of the transactions may be a mixture of New-orders and the other four transactions, although it is to the advantage of the measurement to have as many New-orders as possible.

CPW Metrics

The performance metric of CPW is the relative system throughput for the New-order transaction. This is expressed as the Relative System Performance Metric. Since the other four transactions have minimum frequency requirements and since the New-order transaction is the primary transaction in the benchmark, the throughput is measured in terms of New-orders only. A valid CPW Value is achieved when the correct transaction mix is used and at least 90% of the New-order transactions complete within 5 seconds.

Appendix B. AS/400 Sizing Tools

In this section two V3R6 sizing tools are discussed. For more complete information on AS/400 system sizing tools (prior to V3R6) please refer to *Performance Capabilities Reference Guide, ZC41-8166,* Appendix E.

This section covers:

- the AS/400 Capacity Planner (BEST/1 for the AS/400)

  Best for MES upgrade sizing, or complex ′new business′ system sizing.

- the AS/400 BATCH400 tool,

  Best for MES upgrade sizing where the ′Batch Window′ is important.

B.1 AS/400 CAPACITY PLANNER (BEST/1 for the AS/400)

BEST/1 for the AS/400 is the product of an alliance with BGS Systems, Inc, and will continue to be a part of the IBM Performance Tools/400 product. The capacity planner gives predicted performance information for response times, throughputs, and device utilizations based on estimated and/or measured workloads with a system configuration.

What It Does

The capacity planner helps to analyze the present and future performance requirements for the AS/400 system. The capacity planner allows the use of predefined profiles and/or measured data to create an environment similar to the application environment required.

Use the predefined profiles for an initial proposal. Use the measurement capability alone if the current activity is growing or being analyzed. Mix the predefined profiles with the measured data if new applications are being added or the current ones are being changed significantly. The workloads are then mixed based on the number of local and/or remote devices specified. Optionally, the user can specify a response time or throughput objective for each of the workloads. These objectives (maximum for response time and minimum for throughput) represent the performance requirements.

After the workload has been defined, the capacity planner uses the measured configuration or allows the user to select from an IBM supplied list of configurations. The configuration and workload are analyzed and modeled to predict performance parameters such as response times, throughputs, and device utilizations. When measured configurations are not available, BEST/1 models perspective hardware configurations based on service times measured from a RAMP-C environment.

The capacity planner′s evaluator then compares these numbers against a set of utilization guidelines and the optional response times or throughput objectives. If either the guidelines or objectives are not met, the evaluator recommends an upgrade to the system and reevaluates the adjusted system. This iterative process continues until a configuration is found that satisfies the guidelines and objectives.

Additionally, the planner includes a system growth function. The growth function allows the user to specify an anticipated growth rate over the entire system or by specific workloads. The capacity planner then estimates what configuration changes are required to sustain performance over time.

What Is Supported

The actual AS/400 workload can be measured using the AS/400 Performance Monitor. BEST/1 uses this data to model system activities and provide workload support for the normal AS/400 environment, and also functions such as PC Support (Work Station Feature and Shared Folders) and Display Station Passthrough (Source and Target).

BEST/1 also includes a set of predefined workloads which can be used to represent applications and workloads which are not measured. The predefined profiles include RAMP-C, Officevision/400, RTW, Batch, and Spool.

Support is provided for the various system functions, including checksum protection, purge option, and disk mirroring. In addition, BEST/1 also supports multiple memory pools, multiple priorities, multiple ASPs, batch, batch and interactive relationships, and the ability to model hardware enhancements the day they are announced.

BEST/1 for the AS/400 allows batch job analysis to be based on pool, priority, pathlength, and I/O characteristics the same as can be done for interactive jobs. This allows the user to set objectives for batch throughput, independent of interactive work, or in relation to interactive work.

BEST/1 provides a rated throughput for batch expressed in transactions per hour. This information can be used to estimate changes in throughput based on configuration or workload changes. BEST/1 does not provide detailed batch window analysis or job scheduling analysis. For modeling help in this area, reference B.2, "BATCH400" on page 318.

The capacity planner can also be used to assist the System/36 customer in selecting an appropriately sized IBM AS/400 system to meet their performance requirements. The capacity planner works with a System/36 migration utility procedure which is part of the System/36 release 5.1 coexistence PTF package (DK3700 or later) and System/36 Release 6. The utility uses the S/36 measured performance data created by the System Measurement Facility (SMF), and through modeling, translates the data into AS/400 System/36 environment performance data for the AS/400 Capacity Planner. The capacity planner can then be used to determine an AS/400 configuration that meets the anticipated performance needs.

Where to Get It

This capacity planner is part of the previously mentioned IBM AS/400 Performance Tools package which is a licensed program for the AS/400 system (5763-PT1 in V3R1 and 5716-PT1 in V3R6). This package also includes the measurement facilities needed to use the measurement interface capabilities of the capacity planner. This product′s users guide includes more details on the measurements and capacity planner as well as specifics for the Capacity Planner System/36 Migration Utility option (*IBM AS/400 Programming: Performance Tools Guide*, SC41-8084 and *IBM AS/400 BEST/1 Capacity Planning Tool Guide*, SC41-3341). Other references also include the Capacity Planning

Redbook (GG24-3908), and the BEST/1 Educational Video package (SK2T-6740-00 for 1/2″ open reel, SK2T-6741-00 for 1/2″ cartridge, SK2T-6742-00 for 1/4″ cartridge, and SK2T-6743-00 for 8mm data tapes).

A Skill Dynamics Education course is available. The course is number CEM4930C - ″AS/400 Capacity Planning using BEST/1.″

Announcement PTF summaries for BEST/1

For the latest PTFs that are available to support the AS/400 Advanced Series with PowerPC AS technology, refer to HONE item RTA000089352. If you do not have access to HONE, please contact an IBM representative for this information.

V3R6 Enhancements and PTF Changes for V3R1, V3R0.5, and V2R3

- ***Analysis Enhancements***

  With the October Announcement, BEST/1 now supports the entire range of PowerPC-based Advanced Series AS/400 models.

  Specifically this includes the following systems:

  510 - 2144, 2143
  500 - 2142, 2141, 2140
  400 - 2133, 2132, 2131, 2130

  50S - 2121, 2120
  40S - 2110

  530 - 2153, 2152, 2151, 2150
  53S - 2156, 2155, 2154

  For V3R1, two new ″family″ keywords have been added to the CPU Model Definition:

  – *POWERAS - 510, 500, and 400 models

  – *POWERSRV - 50S and 40S models

  In V3R1, the ″Upgrade to family″ keyword for all CISC-based processors has been intentionally left as *ADVSYS or *ADVSRV. Explicit action is required by the user to configure BEST/1 to automatically trigger upgrade recommendations to PowerPC AS systems by changing the ″Upgrade to family″ keyword to *POWERAS or *POWERSRV on the processors where this is desired.

  In V2R3, there is no ″Upgrade to family″ setting. BEST/1 only upgrades to CPU models which are currently available. As long as Advanced Series and Advanced Servers have their ″Currently available″ keyword set to Y=Yes, other CPU models will upgrade to them, because they appear earlier in the Hardware characteristics menu than the PowerPC AS models (with the exception of the 30S 2411 and 2412 models). If one wants BEST/1 to automatically trigger upgrade recommendations to PowerPC AS systems, one must change the ″Currently available″ keyword of the Advanced Series and/or Advanced Server models to N=No and change the PowerPC AS models to Y=Yes. V2R3 also contains internal rules which allow Server models to only upgrade to and from other Server models. Portables can not be upgraded to or from other CPU models.

BEST/1 now provides the capability to model the transition between CISC-based (Complex Instruction Set Computer) and RISC-based (Reduced Instruction Set Computer) or PowerPC technology based systems. IBM supplied values (not currently user-modifiable) are provided to manage the conversion at a transaction-level for the following values:

- CPU time
- Working Set Size
- I/O counts

These conversion factors are specified for General purpose, CPU intensive, I/O intensive, and Development workloads and are applied during analysis when BEST/1 models are analyzed with a configuration containing a RISC-based CPU model. The current conversion factors are listed in Table 100 below:

Table 100. CISC-to-RISC Conversion Factors

Description	**CPU Time**	**Working Set Size***	**I/O Counts**
General purpose workload type	1.00	2.00	1.00
CPU intensive workloads where the CPU per I/O rate is high	0.50	1.50	1.00
I/O intensive workloads where the CPU per I/O rate is low	1.00	2.30	1.00
Development workload where compile and debug are the bulk of the work	1.50	5.00	1.00

Note:

(*) In addition to the working set size conversion factor, the minimum RISC machine pool requirement of 16 MB (regardless of total system mainstore memory size) is included when recommending pool sizes during a CISC-to-RISC upgrade. In extreme cases this will cause excess memory to be recommended during a growth analysis because BEST/1 preserves the ratio of the machine pool to total mainstore memory as the other pools are increased in size due to workload growth.

DISCLAIMER: CONVERSION FACTORS AND BEST/1 RESULTS HAVE NOT BEEN SUBMITTED TO ANY FORMAL IBM TEST AND ARE DISTRIBUTED ON AN ″AS IS″ BASIS AT THIS TIME WITHOUT ANY WARRANTY EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRENTIES OF MERCHANABILITY AND FITNESS FOR PARTICULAR PURPOSE. The use of BEST/1 results is a customer responsibility and is dependent on the customer′s operational environment; customers applying BEST/1 results do so at their own risk. Conversion factors may change as additional experience is gained by analyzing comparable performance data on CISC and RISC systems. New PTFs will be made available as appropriate.

*NORMAL and *BATCHJOB are the default and recommended settings for workload type and, furthermore, are the only workload types available in V2R3 and V3R0.5. These types will refer to conversion factors which are expected to provide good results on most general purpose customer configurations. The *TRNxxxxx workload types represent specific workload characteristics, and are provided for users who understand how they match the workloads in their model per Table 101 on page 317:

Table 101. Workload Type Characteristics

Description	Normal	Batch
General purpose workload type	*NORMAL	*BATCHJOB
CPU intensive workloads where the CPU per I/O rate is high	*TRNNORM1	*TRNBAT1
I/O intensive workloads where the CPU per I/O rate is low	*TRNNORM2	*TRNBAT2
Development workload where compile and debug are the bulk of the work	*TRNNORM3	*TRNBAT3

Additional description:

- *NORMAL and *BATCHJOB

  These default workloads are represented by traditional commercial applications. CPU profiles for these workloads have up to 10-20% of their CPU time spent in application programs and the remaining 80+% spent in operating system programs. This is because typical RPG and COBOL business applications utilize a significant amount of system services such as database I/O, query processing, workstation/printer processing, and communication I/O.

- *TRNNORM1 and *TRNBAT1

  These types of workloads are referred to as Application Compute Intensive. In these workloads, a majority of the CPU time is is spent in application programs. Although these applications can be written in other languages, they are typically written in in ILE C. Examples of these types of applications are: financial modeling applications which do a significant amount of numeric calculations, statistical analysis applications, 4GL interpreters, and applications which implement complex business rules.

- *TRNNORM2 and *TRNBAT2

  These types of workloads are characterized on a properly configured and tuned system by the majority of application time being spent doing I/O.

- *TRNNORM3 and *TRNBAT3

  These types of workloads are characterized by the OPM development environment where compiles are done with *NOPTIMIZE. Specifying *OPTIMIZE can increase the CPU time. For ILE development environments use the *NORMAL workload type.

To aid in the classification of a workload Table 102 on page 318 below illustrates some sample CPU per I/O values for various DASD response times. One can analyze their BEST/1 workload details to determine if any workload types should be changed. Calculate the CPU per I/O value by dividing the CPU seconds per transaction by the total reads and writes per transaction. For example, if the workload has a single function executing 1 function per user, the function defines a single transaction executing 1 transaction per function, and the transaction specifies 100 I/Os and 5 CPU seconds, then it is characterized by a CPU per I/O value of 0.05 (5 / 100 = 0.05). Thus if the average DASD response time for the configuration is 20 milliseconds this would indicate the *NORMAL classification is the correct setting for the workload.

(Note: if there is more than 1 transaction definition in the workload, one will need to account for the number of transactions in the calculation such that the CPU per I/O value is a weighted average: if 10 transactions have a value of 0.05 and 40 transactions have a value of 0.15, then the weighted average value is 0.13).

Table 102. Sample CPU per I/O values and the Appropriate Workload Type

Avg I/O Resp. Time	**I/O Intensive.** ***TRNNORM2** **or** ***TRNBAT2**	***NORMAL** **or** ***BATCHJOB**	**CPU Intensive** ***TRNNORM1** **or** ***TRNBAT1**
10 msec	< 0.003	0.003 to 0.040	> 0.040
20 msec	< 0.005	0.005 to 0.080	> 0.080
30 msec	< 0.008	0.008 to 0.120	> 0.120

- ***Performance Enhancements for V3R6***

  BEST/1 now runs in ILE instead of EPM, resulting in better performance.

- ***Model Creation Enhancements for V3R6***

  BEST/1 now supports the ADV36 Job Type.

- ***Usability Enhancements for V3R6***

  BEST/1 supports changing the feature of multiple IOP′s, controllers and arms during a single operation.

- ***Predefined Workloads for V3R6***

  Additional server predefined workloads have been added to provide workloads for multimedia environments.

B.2 BATCH400

BATCH400 is a tool for Batch Window Analysis available for V3R6 systems. It is an internal use only tool at this time. Instructions for requesting a copy are at the end of this description.

BATCH400 is a tool to enable AS/400 batch window analysis to be done using information collected by the OS/400 performance monitor.

BATCH400 addresses the often asked question: ′What can I do to my system in order to meet my overnight batch run-time requirements (also known as the Batch Window).′

The BATCH400 tool creates a ′model′ from AS/400 performance data. This model will reside in a file called ′QSBSCHED′ in the target library. The tool can then be asked to analyze the model and provide results for various ′what-if′ conditions. Individual batch job run-time, and overall batch window run-times will be reported by this tool.

BATCH400 Output description:

1. Configuration summary shows the current and modeled hardware for DASD and CPU.
2. Job Statistics show the modeled result followed by the original (probably measured) data for each workload. Workloads are given short names (like b6) that represent either a single batch job or a collection of jobs grouped together. A listing at the end of the output shows the job number/user/name

associated with each workload name. A short name like b6 indicates that a job is in the 6th ′thread′ of jobs, since the letter is ′b′, it is the second job in the thread (a6 being the first). Most other fields in this section are self explanatory. (Tr/Sec) is the Sync I/O per second rate for batch jobs. (Tr/Sec) is the Interactive transaction/second for the interactive workload. (Int Rt) is the Interactive response time. (Bat eff) is the batch efficiency of the workload, a value of 1.0 means the the workload is 100% CPU bound. (ExWait) is the time we had to add to the workload to account for the entire time the job was present in the AS/400, large values here can indicate either delay jobs, or excessive DB contention.

3. Thread summary shows the start/stop/elapsed time for entire threads.

4. Graph of Threads vs. Time of Day shows a ′horizontal′ view of all threads in the model. This output is very handy in showing the relationship of job transitions within threads. It might indicate opportunities to break threads up to allow jobs to start earlier and run in parallel with jobs currently running in a sequential order.

5. Total CPU utilization shows a ′horizontal′ view of how busy the CPU is. This report is on the same time-line as the previous Threads report.

6. Bar chart of Thread elapsed Times shows a comparison of all threads based on end-to-end thread run time. All threads start at time of zero and reach up into this report depending on how long they run. This report shows a ′vertical′ depiction of each thread, and will occasionally show better job transition details than the ′horizontal′ view noted above. This report can help to identify the longest running thread and possible candidate workloads to be improved, or scheduled for different times.

7. The rest of the output shows the model that is stored in QSBSCHED. This is the model that was used for the analysis. The config summary, and workload details are followed by a listing of the workload definitions. This workload definition usually shows:

   - interactive workloads which are a summary of all interactive jobs at a given priority level (type 1 workloads).
   - System workloads which are a summary of various VMC tasks and ′short running′ batch jobs at a given priority level (type 3).
   - Batch workloads which are a summary of individual batch jobs which are usually the focus of the analysis (type 2).
   - Async workloads which are a summary of all the Asynchronous I/O tasks running on the system (type 4).

After looking at the results, use the BATCH400 *CHANGE option to invoke WRKOBJPDM. This will allow you to edit the model. You can alter the job dependencies. Maybe job b6 doesn′t have to follow job a6 in thread 6. You can remove the previous job linkage for b6 and rename b6 job to b12 (where 12 is the next available thread number). Upon saving and exiting, BATCH400 will analyze the new model.

For now this tool is an internal use only tool that can be requested by issuing the following command from a VM command line:

REQUEST BATCH400 FROM DENNEY AT RCHVMV

Appendix C. DASD IOP Device Characteristics

This appendix describes the DASD models supported by the 6502, 6512, 6530, 6532, 6751 and 6754 IOPs and the 2726, 2740, 2741 and 9728 IOAs.

6502/6512 Disk Unit Controller for RAID

Feature #6502 is a disk controller with a 2MB write-cache. Feature #6512 is an enhanced disk controller with a 4MB write-cache. Both can provide RAID-5 protection for up to 16 internal disk units installed in the Storage Expansion Units (#5051/5052). Additionally, disk units attached with #6502 or #6512 and not in a RAID array can be mirrored and/or unprotected. In the RAID configuration, disk unit protection is provided at less cost than mirroring, and with greater performance than system checksums. The 6502 and 6512 also supports mixing different internal disk features on the same controller.

6502/6512 Supported DASD Models

Table 103. 6502/6512 Supported DASD Models

DASD	Model	MB/arm	RAID	Write Cache
6605	050	1031	No	Yes
6605	070	1031	Yes	Yes
6605	072	902	Yes	Yes
6605	074	773	Yes	Yes
6606	050	1967	No	Yes
6606	070	1967	Yes	Yes
6606	072	1721	Yes	Yes
6606	074	1475	Yes	Yes
6607	050	4194	No	Yes
6607	070	4194	Yes	Yes
6607	072	3669	Yes	Yes
6607	074	3145	Yes	Yes
6713	050	8589	No	Yes
6713	070	8589	Yes	Yes
6713	072	7515	Yes	Yes
6713	074	6441	Yes	Yes
6714	050	17548	No	Yes
6714	070	17548	Yes	Yes
6714	072	15354	Yes	Yes
6714	074	13161	Yes	Yes

A minimum of four drives of the same capacity are needed to protect them with RAID-5 protection. A maximum of two arrays are allowed per controller, with a maximum of ten drives allowed per array. All drives in an array must be of the same capacity. Parity is spread on four or eight drives. 1 GB and larger disk units can be RAID-5 protected by the controller. Each System Unit Expansion Tower can support up to 16 disk units on 1 6502/6512 disk controller when the #5052 Storage Expansion Unit is installed. Each DASD Expansion Tower can support up to 32 disk units on 2 6502/6512 disk controllers.

For the Model 400, up to eight 1 GB disk units can be supported when using the #7117 Integrated Expansion Unit.

6530 Storage Device Controller

The #6530 is a Storage controller for up to 16 disk units installed in the Storage Expansion Units (#5051/5052). The 6530 does NOT have a Write Cache and does NOT support RAID. The 6530 also supports mixing different internal disk features on the same controller.

6530 Supported DASD Models

Table 104. 6530 Supported DASD Models

DASD	Model	MB/arm	RAID	Write Cache
6605	030	1031	No	No
6606	030	1967	No	No
6607	030	4194	No	No
6713	030	8589	No	No
6714	030	17548	No	No

Each System Unit Expansion Tower can support up to 16 disk units on 1 6530 disk controller when the #5052 Storage Expansion Unit is installed. Each DASD Expansion Tower can support up to 32 disk units on 2 6530 disk controllers.

6533/6754 Disk Unit Controller for RAID

Feature #6533 (also #6532) is a disk controller with a 4MB write-cache. It can provide RAID-5 protection for up to 16 internal disk units installed in the Storage Expansion Units. Feature #6754 (also #6751) is a Multi-Function IOP with a 4MB write-cache. It can provide RAID-5 protection for up to 20 internal disk units installed in the System Unit. Additionally, disk units attached with the #6533 or #6754 and not in a RAID array can be mirrored and/or unprotected. The 6533 and 6754 also support mixing different internal disk features on the same controller.

6533/6754 Supported DASD Models

Table 105. 6533/6754 Supported DASD Models

DASD	Model	MB/arm	RAID	Write Cache
6606	050	1967	No	Yes
6806	050	1967	No	Yes
6606	070	1967	Yes	Yes
6806	070	1967	Yes	Yes
6606	072	1721	Yes	Yes
6806	072	1721	Yes	Yes
6606	074	1475	Yes	Yes
6806	074	1475	Yes	Yes
6607	050	4194	No	Yes
6807	050	4194	No	Yes
6607	070	4194	Yes	Yes
6807	070	4194	Yes	Yes
6607	072	3669	Yes	Yes
6807	072	3669	Yes	Yes
6607	074	3145	Yes	Yes
6807	074	3145	Yes	Yes
6713	050	8589	No	Yes
6813	050	8589	No	Yes
6713	070	8589	Yes	Yes
6813	070	8589	Yes	Yes
6713	072	7515	Yes	Yes
6813	072	7515	Yes	Yes
6713	074	6441	Yes	Yes
6813	074	6441	Yes	Yes
6714	050	17548	No	Yes
6714	070	17548	Yes	Yes
6714	072	15354	Yes	Yes
6714	074	13161	Yes	Yes

Note: 6806, 6807, 6813 and 6714 are Ultra-SCSI DASD.

A minimum of four drives of the same capacity are needed to protect them with RAID-5 protection. A maximum of two arrays are allowed per controller, with a maximum of ten drives allowed per array. All drives in an array must be of the same capacity. Parity is spread on four or eight drives. Each System Unit Expansion Tower can support up to 16 disk units on 1 6533 disk controller. Each DASD Expansion Tower can support up to 32 disk units on 2 6533 disk controllers.

For the Models 640 and 650, up to 20 disk units can be supported on 1 6754 MFIOP.

2741/2740/2726 Disk Unit Controller for RAID

Feature #2741 (also #2726) is a disk controller with a 4MB write-cache. It can provide RAID-5 protection for up to 15 internal disk units installed in the PCI System Unit or PCI Expansion Unit. Feature #2740 is a low cost disk controller with a 4MB write-cache that is targeted for smaller systems. It can provide RAID-5 protection for up to 10 internal disk units installed in the PCI System Unit or PCI Expansion Unit. Additionally, disk units attached with the #2741 or #2740 and not in a RAID array can be mirrored and/or unprotected. The 2741 and 2740 also support mixing different internal disk features on the same controller.

2741/2740/2726 Supported DASD Models

Table 106. 2741/2740/2726 Supported DASD Models

DASD	Model	MB/arm	RAID	Write Cache
6606	050	1967	No	Yes
6806	050	1967	No	Yes
6606	070	1967	Yes	Yes
6806	070	1967	Yes	Yes
6606	072	1721	Yes	Yes
6806	072	1721	Yes	Yes
6606	074	1475	Yes	Yes
6806	074	1475	Yes	Yes
6607	050	4194	No	Yes
6807	050	4194	No	Yes
6607	070	4194	Yes	Yes
6807	070	4194	Yes	Yes
6607	072	3669	Yes	Yes
6807	072	3669	Yes	Yes
6607	074	3145	Yes	Yes
6807	074	3145	Yes	Yes
6713	050	8589	No	Yes
6813	050	8589	No	Yes
6713	070	8589	Yes	Yes
6813	070	8589	Yes	Yes
6713	072	7515	Yes	Yes
6813	072	7515	Yes	Yes
6713	074	6441	Yes	Yes
6813	074	6441	Yes	Yes
6714	050	17548	No	Yes
6714	070	17548	Yes	Yes
6714	072	15354	Yes	Yes
6714	074	13161	Yes	Yes

Note: 6806, 6807, 6813 and 6714 are Ultra-SCSI DASD.

A minimum of four drives of the same capacity are needed to protect them with RAID-5 protection. A maximum of two arrays are allowed per controller, with a maximum of ten drives allowed per array. All drives in an array must be of the same capacity. Parity is spread on four or eight drives.

9728 Storage Device Controller

The #9728 is a Storage controller for up to 5 disk units installed in the PCI System Unit. The 9728 does NOT have a Write Cache and does NOT support RAID. The 9728 also supports mixing different internal disk features on the same controller.

9728 Supported DASD Models

Table 107. 9728 Supported DASD Models

DASD	Model	MB/arm	RAID	Write Cache
6606	030	1967	No	Yes
6806	030	1967	No	Yes
6607	030	4194	No	Yes
6807	030	4194	No	Yes
6713	030	8589	No	Yes
6813	030	8589	No	Yes
6714	030	17548	No	No

Note: 6806, 6807, 6813 and 6714 are Ultra-SCSI DASD.