DB2 Family: Client/Server Performance Measurement Series
DB2/2 <> DB2/CAE/Windows

Document Number Version 1.2

June 26, 1996

Robert Gilles
gilles@vnet.ibm.com
GILLES at STLVM14

Todd Munk
munk@stlps.vnet.ibm.com
MUNK at STLVM14

Hugh Smith
hugh@vnet.ibm.com
SMITHHJ at STLVM14

DB2 Performance
Santa Teresa Laboratory
San Jose, California

© **Copyright International Business Machines Corporation 1996. All rights reserved.**
Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Trademarks

The following are trademarks of the IBM Corporation. An asterisk indicates the trademark is registered.

- DatagLANce
- DB2 *
- DB2/2
- IBM *
- LANStreamer
- OS/2 *
- PS/2 *

All other trademarks are the properties of their respective owners.

© Copyright IBM Corp. 1996

Disclaimer

References in this technical report 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 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.

The information contained in this technical report 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.

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

© Copyright IBM Corp. 1996

Acknowledgements

The following people helped make this paper possible.

- Jerry Heglar
- Serge Limoges
- Mark Ryan
- Bill Wilkins

© Copyright IBM Corp. 1996

Contents

© Copyright IBM Corp. 1996

1. Introduction

This technical report is one of a series of technical reports dealing with performance prediction and capacity planning issues for the DB2 family of products operating in a client/server configuration. The goal of these technical reports is to help the customer:

- Assess proposed configurations for acceptable performance
- Assess performance of current configurations
- Choose proper hardware capacity for a proposed configuration
- Choose the proper software programs for a proposed configuration
- Tune a new or current configuration

This report is unique because it describes the impacts on the numerous components involved in a client/server environment as the workload increases in that environment. Use this report as an example configuration that could be analyzed and potentially applied to other configurations. This report describes the throughput, CPU utilization, RAM utilization, network utilization and response times when PowerBuilder clients and OS/2 clients call stored procedures on DB2 to perform OLTP type transactions. Use the results from these measurements to predict performance and plan for capacity in your environment. Particularly noteworthy items are highlighted as **general considerations** in this report.

© Copyright IBM Corp. 1996

2. Configuration

This technical report provides information in regards to the following configuration:

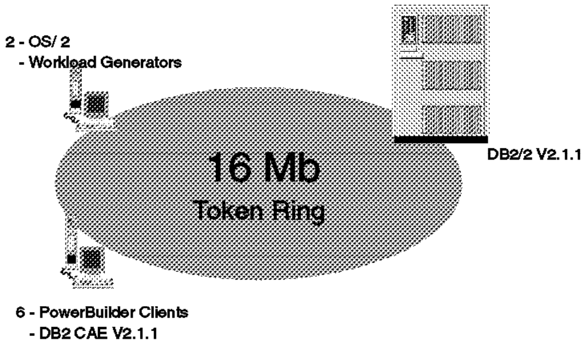

Figure 1. Configuration with DB2 for OS/2

Specifications

- **DB2 for OS/2** Version 2.1.1. The processor is a PC500 server with a 90 Mhz Pentium processor and 128 MBytes of memory. The PC500 is attached to a 16 Mbit LAN using a LANStreamer adapter card and configured to use early token release.
- **6 - PowerBuilder Version 4.0.2 with DB2 CAE for Windows Version 2.1.1** clients running on 9595 OPT PS/2s with 60 Mhz Pentium Processors and 80 MBytes of memory. The LAN adapter cards are configured to use early token release.
- **2 - 9595 OPT PS/2s** with 60 Mhz Pentium Processors running OS/2 Version 3.0 (WARP). These PS/2s are used to increase the workload in the configuration.
- **16 Mbit Token Ring LAN**

Network protocols in use between the components are as follows:

- TCP/IP is in use between the PowerBuilder clients and DB2 for OS/2

© Copyright IBM Corp. 1996

- TCP/IP is in use between the OS/2 clients and DB2 for OS/2

The components in the configuration are monitored using the following programs:

- DB2 for OS/2: Database System Monitor Snapshot
- OS/2: System Performance Monitor/2 (SPM/2)
- Clients: Internal tools to measure elapsed time and transactions per second
- LAN: DatagLANce Network Analyzer for Ethernet and Token Ring for OS/2

3. Workload Description

The IBM Relational Warehouse Workload (IRWW) is used for the measurements. IRWW is a simulated product warehouse, customer order, and order delivery system.

A series of measurements are performed, with each subsequent run providing an increased workload to the configuration. All measurements include six PowerBuilder clients running a predefined set of transactions in random order without modification for the entire series of measurements. The PowerBuilder clients provide the ″end user″ perspective of the effects of increasing the workload in the configuration. The workload is increased each measurement by increasing the number of sessions on the two OS/2 workload generators. The OS/2 workload generators run the exact same transaction mix as the six PowerBuilder clients. The only difference is the management of screen I/O at the client. The PowerBuilder clients simulate a real application, where the OS/2 workload generators do not perform screen I/O.

Transaction Characteristics

The workload consists of seven transactions. Each transaction consists of a stored procedure call followed by a commit or rollback. This results in two ″round trip″ network flows between the client and the server. The stored procedures are written in ″C″ and they perform seven distinct functions in a predefined mix. The seven transactions, a brief description, and the percentage of the mix are:

- Transaction **Tx1**

  Performs various selects, fetches, updates, and inserts in support of the receipt of new customer orders. Transaction Tx1 runs as twenty-two percent of the total transaction mix and is the most complicated transaction.

- Transaction **Tx2**

  Performs various selects and fetches in support of providing the status of an order. Transaction Tx2 runs as twenty-four percent of the total transaction mix.

- Transaction **Tx3**

  Performs various selects, fetches, updates, and inserts in support of received customer payments. Transaction Tx3 runs as twenty-two percent of the total transaction mix.

- Transaction **Tx4**

  Performs an update in support of changing the price of an item. Transaction Tx4 runs as one percent of the total transaction mix.

- Transaction **Tx5**

  Performs various selects in support of providing the price of a set of items. Transaction Tx5 runs as twenty-five percent of the total transaction mix.

- Transaction **Tx6**

  Performs a join and various selects in support of providing the current stock level of an item. Transaction Tx6 runs as four percent of the total transaction mix.

- Transaction **Tx7**

© Copyright IBM Corp. 1996

Performs various selects, updates, and deletes in support of the delivery of a group of orders. Transaction Tx7 runs as two percent of the total transaction mix.

These transactions are described in greater detail later in this report.

Table Characteristics

Before each measurement, the tables on DB2 for OS/2 are put into a consistent state as defined in Table 1. The table shows the initial number of rows at the start of a measurement, whether the number of rows is fixed or will grow throughout the measurement, the number of columns, the number of indexes, and the preload into buffer percentage. The amount of preloaded data is greater than available buffer space, so table T9 is loaded first, followed by table T8, then T7, on up to T2. DB2 will remove portions of the earlier preloaded tables as space is needed. The transactions manipulate these tables in various ways and they are designed to have a high degree of contention on some of the tables, such as the T2 table.

Table 1. Table Characteristics

Table	Initial Rows (length)	Row Count?	Columns	Indexes	PreLoad
T1	0 (Fixed 52)	Variable	8	0	0
T2	8 (Fixed 99)	Fixed	9	1	100
T3	80 (Fixed 105)	Fixed	11	1	100
T4	72,000 (Fixed 8)	Variable	3	1	12
T5	100,000 (Fixed 86)	Fixed	4	1	100
T6	240,000 (Fixed 25)	Variable	8	1	12
T7	240,000 (Variable, 675 max)	Fixed	21	2	12
T8	800,000 (Fixed 306)	Fixed	17	1	20
T9	2,399,791 (Fixed 60)	Variable	10	1	12

Locking

Locking is system defined and managed. All packages are bound with an isolation level of cursor stability.

Physical location

There are four disk drives. One drive contains the DB2 code and log files. The other three disk drives are included in a single database managed tablespace.

For general consideration, it is best to allow the database to manage the tablespaces.

Buffer Pool Allocation

Two separate measurement sets are described in this report. One measurement set defines a buffer pool of 16384 (64 MByte) and the other measurement set defines a buffer pool of 12288 (48 MByte). A comparison of both measurement sets provides information for memory impact analysis.

4. Results

The next sections describe the results from the measurements. The effects on each component in the configuration are described as well as the performance of each transaction. The component results can help you understand hardware capacity issues. The transaction results can help you understand the perceived performance prediction issues.

The results are presented in relation to the overall throughput achieved at each workload level. This is described as transactions per second (TPS). As the workload increased, the throughput increased. The resultant effects on each component is then presented based on the current achieved throughput. For example, when achieving 8.4 transactions per second:

- the PC500 running DB2 for OS/2 is 92.8 percent utilized
- the traffic on the 16 Mbit LAN is 25,254 bytes per second and the LAN is 1.26 percent utilized
- etc...

The process of tuning the application and database is ongoing. These measurements achieved a throughput rate of 9.05 transactions per second. At this point, the CPU is fully utilized and is identified as the primary constraint preventing additional throughput.

Client Equivalence

The configuration includes six PowerBuilder clients and a variable number of OS/2 clients. All clients run the same transaction mix, but the PowerBuilder clients present the output data on the computer screen. This adds to the response time, which decreases the throughput for the PowerBuilder clients. This section estimates a rough number of total PowerBuilder clients based on the throughput obtained by the OS/2 clients plus the throughput of the actual PowerBuilder clients.

Table 2 on page 9 describes the number of clients activated to achieve the throughput defined in column two. Column three describes the number of PowerBuilder clients. This is always 6. Column four describes the number of OS/2 clients. This is increased for each measurement. Column one is the actual number of clients, which is the sum of columns three and four. In parenthesis is the buffer pool size for that measurement. The measurements with a 48 MByte buffer pool achieved slightly less throughput for each equivalent measurement using a 64 MByte buffer pool. Column five is the estimated number of PowerBuilder clients. This column is derived by evaluating the number of equivalent PowerBuilder clients each OS/2 client represents in terms of throughput. The equation for evaluating this is:

$$\text{Estimated PowerBuilder Clients} = \left[\frac{\text{OS/2 client tps}}{\text{PowerBuilder client tps}} * (\text{\# OS/2 clients}) \right] + (\text{\# PowerBuilder clients})$$

```
If
 OS/2 client tps         = .92 tps
 PowerBuilder client tps = .13 tps
 # OS/2 clients          = 9
 # PowerBuilder clients  = 6
then
```

$$\text{Estimated PowerBuilder Clients} = \left[\frac{.92\ \text{tps}}{.13\ \text{tps}} * 9\ \text{clients} \right] + (6\ \text{clients})$$

$$= 69\ \text{clients (approximate)}$$

Table 2. Number of clients in relation to Aggregate Throughput (TPS)

Actual Clients (Buffer Pools Size)	**Throughput (TPS)**	**PowerBuilder Clients**	**OS/2 Clients**	**Est. PowerBuilder Clients**
8 (64 MByte)	4.20	6	2	30
8 (48 MByte)	4.13	6	2	28
9 (64 MByte)	5.76	6	3	40
9 (48 MByte)	Not measured			
10 (64 MByte)	7.04	6	4	49
10 (48 MByte)	6.36	6	4	46
11 (64 MByte)	7.92	6	5	57
11 (48 MByte)	Not measured			
12 (64 MByte)	8.40	6	6	61
12 (48 MByte)	7.81	6	6	58
13 (64 MByte)	8.83	6	7	66
13 (48 MByte)	8.20	6	7	61
14 (64 MByte)	8.9	6	8	67
14 (48 MByte)	8.36	6	8	64
15 (64 MByte)	9.05	6	9	69
15 (48 MByte)	Not measured			

The values obtained in these measurements are obtained based on the actual number of clients as reflected in column one. When evaluating the estimated number of PowerBuilder clients, other capacity planning factors need to be addressed. For example, each client that accesses DB2 for OS/2 (for this type of transaction mix) should be allowed approximately 0.9 MBytes of RAM. These factors need to be taken into consideration when designing a configuration. One last point on the number of clients, these numbers are based on the clients running without think time or keystroke time. So if these aspects are factored in, the actual number of clients could be increased to fill the capacity left available due to the think time and keystroke time.

CPU Utilization

This section describes CPU utilization and RAM utilization as the workload increases for each measurement. Table 3 on page 10 describes the CPU utilization for the measurements with a buffer pool size of 64 MBytes. Figure 2 on page 10 compares the CPU utilization for the measurements with a 64 MByte buffer pool (solid line) and the 48 MByte buffer pool (dotted line). The CPU utilization is slightly higher for the 48 MByte buffer pool (dotted line). Table 4 on page 11 describes the RAM utilization for DB2 for OS/2 for the 64 MByte buffer pool measurement. Table 5 on page 11 describes the RAM utilization for the 48 MByte buffer pool measurement.

At a throughput rate of 9.05 transactions per second, the PC500 reaches 99.56% CPU utilization.

Table 3. CPU utilization in relation to TPS

TPS	PC500 Running OS/2
4.20	46.22%
5.76	61.36%
7.04	75.21%
7.92	87.35%
8.40	92.80%
8.83	96.70%
8.91	98.75%
9.05	99.56%

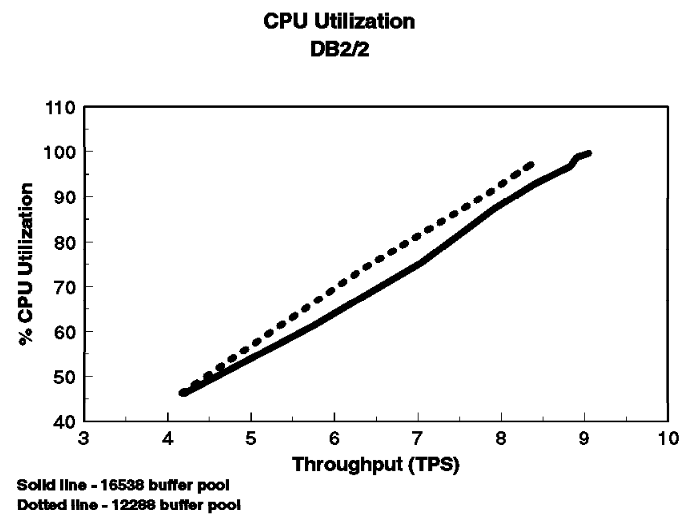

Figure 2. CPU Utilization in Relation to Throughput

Table 4 on page 11 and Table 5 on page 11 show the increase in RAM utilization for the PC500 as the number of clients increases. The initial value is an indication of the RAM requirements when allocating the defined buffer pool sizes. Table 4 on page 11 describes a buffer pool of 64 MBytes and Table 5 on page 11 describes a buffer pool of 48 MBytes. The average amount of RAM used per client is approximately 0.9 MBytes.

Table 4. RAM utilization on PC500 with 64 MByte buffer pool

Number of Clients	MBytes of RAM
8	108.521
9	109.245
10	110.453
11	111.381
12	111.904
13	112.712
14	113.942
15	114.754

Table 5. RAM utilization on PC500 with 48 MByte buffer pool

Number of Clients	MBytes of RAM
8	92.121
10	93.849
12	95.412
13	96.303
14	97.377

For general consideration:

- **DB2 for OS/2 should run on a fast processor. Allow for 0.9 MBytes of memory per client for this type of transaction mix.**
- **Buffer pool allocation has an effect on CPU utilization and throughput. The larger the buffer pool, the less CPU demand and the greater the throughput. Make sure the buffer pool is not set to a value that paging occurs when adding additional clients or some other system task occurs. Paging is a serious performance degradation.**
- **There should be multiple hard disks to allow for spreading the data out to multiple drives. Allow the database to manage the containers set up for the data.**

Network

The network plays a big part in how well the client/server application will perform, and how big an effect the client/server applications will have on the performance of the database. Delay is the biggest enemy. How big a role delay plays, is dependent on the application. Every time a trip across the network is required, if locks are held, they are held for the time it takes for the messages to move between the client and the database. For these measurements, stored procedures are used, which has the benefit of decreasing the number of SQL calls that must flow across the network as messages. Locks are only held for the time to process the stored procedure and for one network round trip to issue a commit or rollback. So although delay does play a role in these measurements, it could play a larger role in other environments.

The next section describes the effects on the LAN during the measurement runs. For this mix of transactions, the message lengths are never greater than 1,200 bytes, so the following values are set:

- 4K RQRIOBLK definitions in the OS/2 clients
- 4K DOS_RQRIOBLK definitions in the DOS clients

LAN

Table 6 shows the percentage of LAN utilization and byte rate as the workload is increased. For these measurements, the LAN capacity does not factor in. The 16 Mbit LAN is lightly utilized. Typically, a production environment will have more activity on the LAN. Use these numbers to analyze the additional load introduced on the LAN due to the mix of transactions used in this configuration.

For general consideration, monitor your LAN to make sure it is not being over utilized. Over utilization of the LAN could introduce delays with resultant bad effects on database performance.

Table 6. LAN utilization in relation to TPS

Throughput (TPS)	Bytes Per Second	% LAN Utilization
4.20	16,657	.83
5.76	20,248	1.01
7.04	23,568	1.18
7.92	26,379	1.32
8.40	25,254	1.26
8.83	28,065	1.40
8.90	28,188	1.41
9.05	27,573	1.38

Transactions

The next section describes the transactions and the achieved response time at the PowerBuilder clients in relation to the aggregate throughput. The response time duration starts when the end user submits the transaction request. This is after the network connection is established and the end user has entered the input data. The duration ends after the commit or rollback has been issued and all the response data is available to the end user (visible on the computer display).

In this section, graphs depict the point where 90 percent of the transactions complete within the identified maximum response time for the identified throughput. Since this is the 90th percentile, this is an indication of the greatest response time when sampling the fastest 90 percent of the transactions. Most transactions achieve a better response time. The solid line is for the measurements with the buffer pool set to 64 MByte, and the dotted line is for a buffer pool of 48 MBytes. The 48 MByte buffer pool measurements achieve a longer response time than the 64 MByte buffer pool measurements. Figure 3 on page 13 displays the aggregate response time of the transactions for the identified throughput.

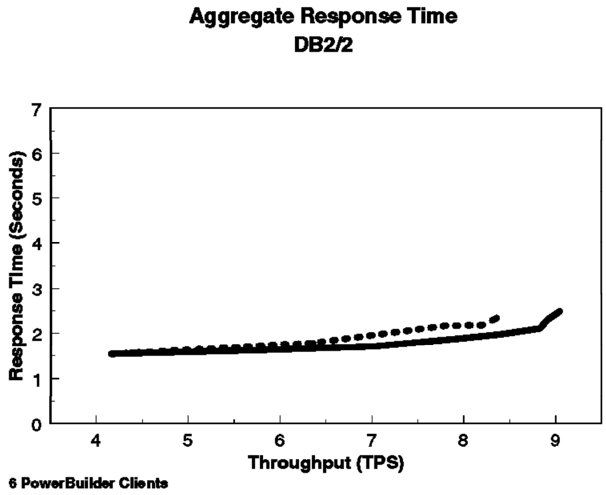

Figure 3. Aggregate Response Time

Each procedure described in the following sections, perform different operations which creates various degrees of contention on the tables. The SQL calls in the order they are called, are described for each procedure. CPU constraint on the PC500 is the primary factor for the response time degradation.

For general consideration:

- **If the procedure is going to be called frequently, a performance benefit can be obtained by adding a SQLZ_HOLD_PROC to the RETURN statement in the procedure. This statement holds the server library in main memory in preparation for the next invocation of the procedure. This does place a burden on memory resources, though.**
- **For OLTP type transactions, a performance benefit can be obtained by specifying KEEPDARI YES in the database manager configuration. This eliminates the DARI setup costs for each new transaction. This does place a burden on memory resources, though.**

Transaction Tx1

Transaction Tx1 is the most involved of the seven transactions. The transaction calls a stored procedure which performs write operations against tables T3, T4, T6, T8, and T9. The stored procedure performs read operations against tables T2, T3, T5, T7, and T8. Index matching predicates in the WHERE clauses optimize access to all tables. The stored procedure has two input parameters for a byte count between 22 and 190. There are nine output parameters with a maximum byte count of 713.

A summary of the SQL calls in the stored procedure and the order in which they occur are as follows:

```
Select
   T7
   T2
Fetch and Update
   T3
Insert
   T4
   T6
Fetch
   T5
Loop up to 15 times
   Fetch and Update
      T8
   Insert
      T9
End loop
```

Figure 4 describes the achieved response time for transaction Tx1. The vertical axis represents the response time as seen by the PowerBuilder clients. The horizontal axis represents the overall throughput rate for the aggregate of transactions. The lines represent the point where 90 percent of the Tx1 transactions achieved this response time or better. The solid line is for the measurements with a 64 MByte buffer pool. The dotted line is for the measurements with a 48 MByte buffer pool.

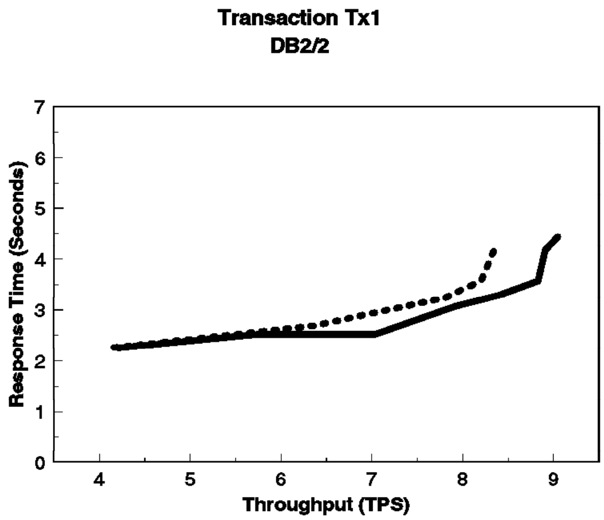

Figure 4. Response Time For Transaction Tx1

The maximum response time for 90 percent of the Tx1 transactions is between 2.0 and 4.5 seconds. Transaction Tx1 shows a gradual response time degradation as the workload increases and an increased degradation at 8 transactions per second for the 48 MByte buffer pool measurements and at 9 transactions per second for the 64 MByte buffer pool measurements. The increased degradation is attributed to CPU constraint on the PC500. The gradual response time degradation can be attributed to the increased workload and resultant contention. Table 7 on page 15 displays the transactions that have potential lock contention with transaction Tx1 on the defined tables.

Table 7. Potential Lock Contention For Transaction Tx1

	Tx1	Tx2	Tx3	Tx4	Tx5	Tx6	Tx7
T2			yes				
T3	yes		yes				
T4	yes						yes
T5				yes			
T6	yes						yes
T7			yes				yes
T8	yes						
T9	yes						yes

Transaction Tx2

Transaction Tx2 calls a stored procedure which performs read operations against tables T6, T7, and T9. Index matching predicates in the WHERE clause optimize access to these tables. The stored procedure has one input parameter for 26 bytes and four output parameters with a maximum byte count of 516.

A summary of the SQL calls in the stored procedure and the order in which they occur are as follows:

```
For 60% of the time:
    Select
       T7
    Loop up to 4 times
       Fetch from
          T7
    End loop
End 60% of the time
    Select
       T7
       T6
    Loop up to 15 times
       Fetch
          T9
    End loop
```

Figure 5 on page 16 describes the achieved response time for transaction Tx2. The vertical axis represents the response time as seen by the PowerBuilder clients. The horizontal axis represents the overall throughput rate for the aggregate of transactions. The lines represent the point where 90 percent of the Tx2 transactions achieved this response time or better. The solid line is for the measurements with a 64 MByte buffer pool. The dotted line is for the measurements with a 48 MByte buffer pool.

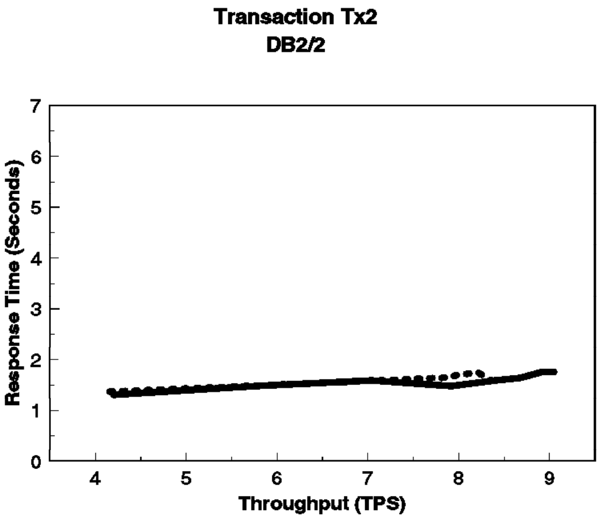

Figure 5. Response Time For Transaction Tx2

The maximum response time for 90 percent of the Tx2 transactions is between 1.0 and 1.5 seconds. Transaction Tx2 begins to show gradual response time degradation throughout the measurements. There is almost no difference between the 48 MByte buffer pool measurements and the 64 MByte buffer pool measurements. The gradual response time degradation can be attributed to increased workload and resultant contention. Table 8 displays the transactions that have potential lock contention with transaction Tx2 on the defined tables.

Table 8. Potential Lock Contention For Transaction Tx2

	Tx1	Tx2	Tx3	Tx4	Tx5	Tx6	Tx7
T6	yes						yes
T7			yes				yes
T9	yes						yes

Transaction Tx3

Transaction Tx3 calls a stored procedure which performs write operations against tables T1, T2, T3, and T7. The stored procedure performs read operations against tables T2, T3, and T7. Access to all tables, except T1, use index matching predicates in the WHERE clauses to optimize access to the tables. The stored procedure has one input parameter for 39 bytes and six output parameters with a maximum byte count of 551.

A summary of the SQL calls in the stored procedure and the order in which they occur are as follows:

```
For 60% of the time
      Select
         T7
      Loop up to 4 times
         Fetch from
            T7
      End loop
End 60% of the time
      Fetch and Update
         T7
      Conditionally Fetch and Update
         T2
         T3
      Insert
         T1
```

Figure 6 describes the achieved response time for transaction Tx3. The vertical axis represents the response time as seen by the PowerBuilder clients. The horizontal axis represents the overall throughput rate for the aggregate of transactions. The lines represent the point where 90 percent of the Tx3 transactions achieved this response time or better. The solid line is for the measurements with a 64 MByte buffer pool. The dotted line is for the measurements with a 48 MByte buffer pool.

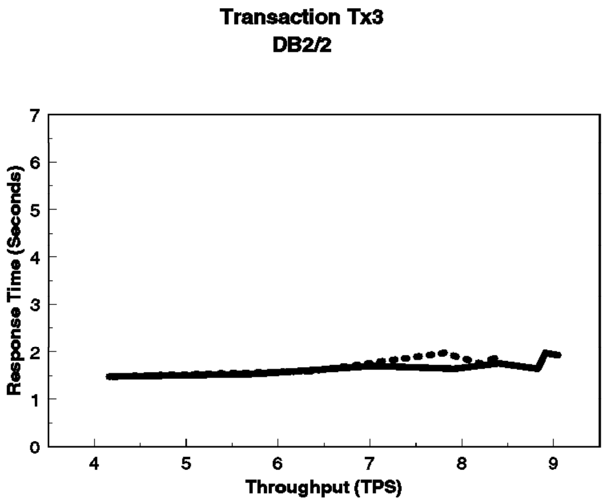

Figure 6. Response Time For Transaction Tx3

The maximum response time for 90 percent of the Tx3 transactions is between 1.5 and 2.0 seconds. Transaction Tx3 shows almost no response time degradation as the workload increases. Table 9 on page 18 displays the transactions that have potential lock contention with transaction Tx3 on the defined tables.

Table 9. Potential Lock Contention For Transaction Tx3

	Tx1	Tx2	Tx3	Tx4	Tx5	Tx6	Tx7
T1			yes				
T2			yes				
T3	yes		yes				
T7			yes				yes

Transaction Tx4

Transaction Tx4 calls a stored procedure which performs a write operation against T5. Index matching predicates in the WHERE clause optimize access to this table. The stored procedure has one input parameter for 8 bytes and three output parameters with a maximum byte count of 99.

A summary of the SQL calls in the stored procedure and the order in which they occur are as follows:

```
Update
  T5
```

Figure 7 describes the achieved response time for this transaction. The vertical axis represents the response time as seen by the PowerBuilder clients. The horizontal axis represents the overall throughput rate for the aggregate of transactions. The lines represent the point where 90 percent of the Tx4 transactions achieved this response time or better. The solid line is for the measurements with a 64 MByte buffer pool. The dotted line is for the measurements with a 48 MByte buffer pool.

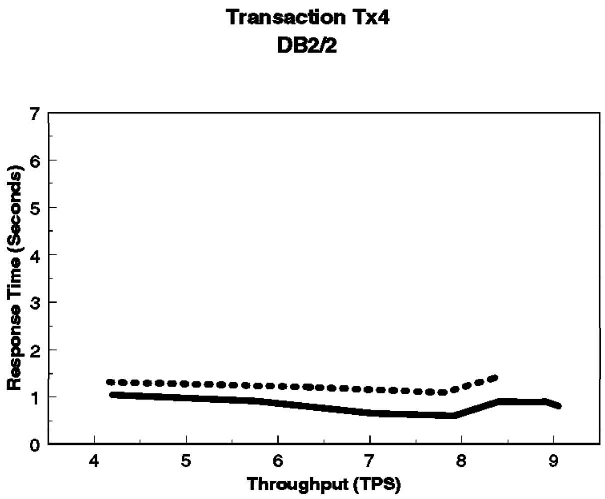

Figure 7. Response Time for Transaction Tx4

The maximum response time for 90 percent of the Tx4 transactions is between .5 and 1.5 seconds. Transaction Tx4 shows a consistent response time with some improvement as the workload increases. Transaction Tx4 updates table 5 which is 100% preloaded into buffers. The response times reflect this. The response times for the 48 MByte buffer pool measurements are consistently higher than the 64 MByte buffer pool measurements. Table 10 displays the transactions that have potential lock contention with transaction Tx4 on the defined tables.

Table 10. Potential Lock Contention For Transaction Tx4

	Tx1	**Tx2**	**Tx3**	**Tx4**	**Tx5**	**Tx6**	**Tx7**
T5				yes			

Transaction Tx5

Transaction Tx5 calls a stored procedure which performs read operations against the T5 table. Index matching predicates in the WHERE clause optimize access to this table. The stored procedure has one input parameter with a byte count between 8 and 120. It has seven output parameters with a maximum byte count of 590.

A summary of the SQL calls in the stored procedure and the order in which they occur are as follows:

```
Loop up to 15 times
   Select
      T5
End loop
```

Figure 8 on page 20 describes the achieved response time for this transaction. The vertical axis represents the response time as seen by the PowerBuilder clients. The horizontal axis represents the overall throughput rate for the aggregate of transactions. The lines represent the point where 90 percent of the Tx5 transactions achieved this response time or better. The solid line is for the measurements with a 64 MByte buffer pool. The dotted line is for the measurements with a 48 MByte buffer pool.

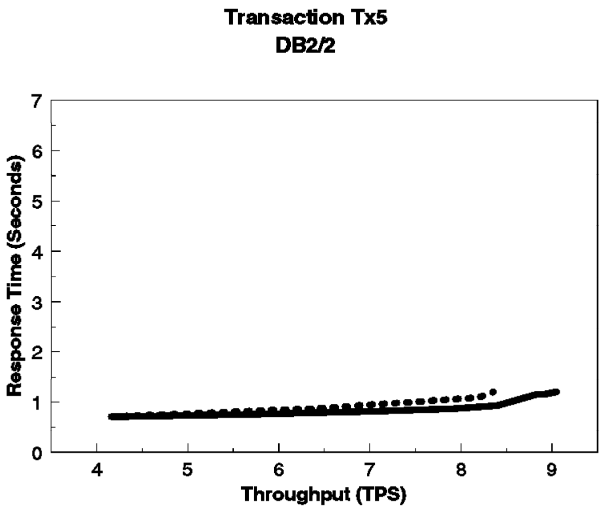

Figure 8. Response Time of Transaction Tx5

The maximum response time for 90 percent of the Tx5 transactions is between .5 and 1.5 seconds. Transaction Tx5 shows a gradual response time degradation as the workload increases. The gradual response time degradation can be attributed to the increased workload and resultant contention. There is little difference between the 48 MByte buffer pool measurements and the 64 MByte buffer pool measurements. Table 11 displays the transactions that have potential lock contention with transaction Tx5 on the defined tables.

Table 11. Potential Lock Contention For Transaction Tx5

	Tx1	Tx2	Tx3	Tx4	Tx5	Tx6	Tx7
T5				yes			

Transaction Tx6

Transaction Tx6 calls a stored procedure which performs read operations against tables T3, T8, and T9. Index matching predicates in the WHERE clause optimize access to the rows. The stored procedure has one input parameter with a byte count of 8 and three output parameters with a maximum byte count of 78.

A summary of the SQL calls in the stored procedure and the order in which they occur are as follows:

```
Select
   T3
Select
   T8 joined with T9
```

Figure 9 on page 21 describes the achieved response time for this transaction. The vertical axis represents the response time as seen by the PowerBuilder clients. The horizontal axis represents the overall throughput rate for the aggregate of transactions. The lines represent the point where 90 percent of the Tx6 transactions achieved this response time or better. The solid line is for the measurements with a 64 MByte buffer pool. The dotted line is for the measurements with a 48 MByte buffer pool.

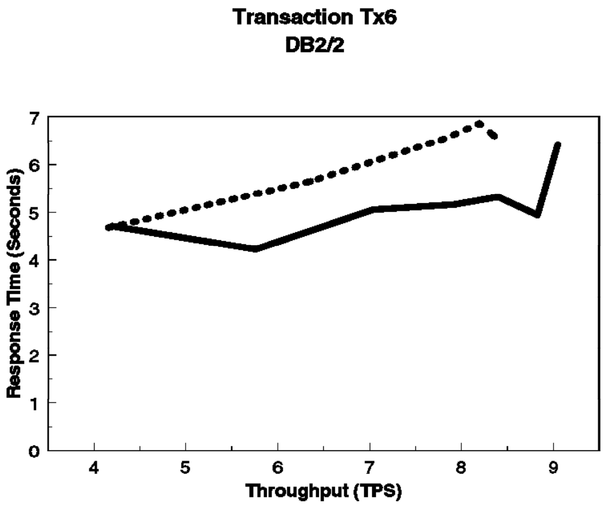

Figure 9. Response Time for Transaction Tx6

The maximum response time for 90 percent of the Tx6 transactions is between 4.0 and 7.0 seconds. Transaction Tx6 shows a response time degradation throughout the measurements. Tx6 is CPU intensive due to the join between tables T8 and T9, so as the CPU demands increase, Tx6 is effected. Add to this the resultant contention on the tables, especially table T3, which is small, and the effect on response time is enhanced. The 48 MByte buffer pool measurements are consistently worse than the 64 MByte buffer pool measurements. Table 12 displays the transactions that have potential lock contention with transaction Tx6 on the defined tables.

Table 12. Potential Lock Contention For Transaction Tx6

	Tx1	**Tx2**	**Tx3**	**Tx4**	**Tx5**	**Tx6**	**Tx7**
T3	yes		yes				
T8	yes						
T9	yes						yes

Transaction Tx7

Transaction Tx7 calls a stored procedure which performs write operations against tables T4, T6, T7, and T9. The stored procedure performs read operations against tables T4, T6, and T9. Index matching predicates in the WHERE clauses optimize access to these large tables. The stored procedure has one input parameter for 6 bytes and three output parameters with a maximum byte count of 78.

A summary of the SQL calls in the stored procedure and the order in which they occur are as follows:

```
 Loop up to 10 times
    Fetch and Delete
       T4
    Fetch and Update
       T6
    Update
       T9
    Select
       T9
    Update
       T7
End loop
```

Figure 10 describes the achieved response time for this transaction. The vertical axis represents the response time as seen by the PowerBuilder clients. The horizontal axis represents the overall throughput rate for the aggregate of transactions. The lines represent the point where 90 percent of the Tx7 transactions achieved this response time or better. The solid line is for the measurements with a 64 MByte buffer pool. The dotted line is for the measurements with a 48 MByte buffer pool.

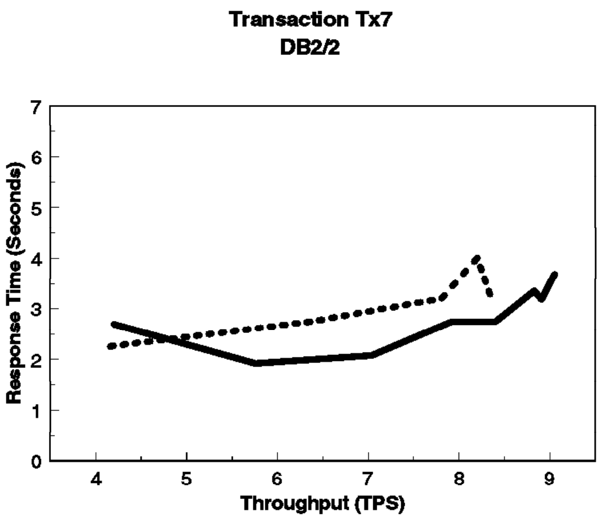

Figure 10. Response Time For Transaction Tx7

The maximum response time for 90 percent of the Tx7 transactions is between 2.0 and 4.0 seconds. Generally, Tx7 shows a gradual response time degradation throughout the scope of the measurements. Since transaction Tx7 needs exclusive locks to tables T4, T6, T7, and T9 tables, as the throughput increases, the response time should degrade. These are all large tables, so the contention should not be drastic. The 48 MByte buffer pool measurements are consistently higher than the 64 MByte buffer pool measurements.

Table 13 displays the transactions that have potential lock contention with transaction Tx7 on the defined tables.

Table 13. Potential Lock Contention For Transaction Tx7

	Tx1	**Tx2**	**Tx3**	**Tx4**	**Tx5**	**Tx6**	**Tx7**
T4	yes						yes
T6	yes						yes
T7			yes				yes
T9	yes						yes

5. Conclusions

This technical report shows the results and analysis of some performance measurements for a client/server relational database configuration that includes DB2 for OS/2 and DB2/CAE/Windows with PowerBuilder. The goal of these measurements is to provide an example environment and analysis that will help a customer:

- Assess proposed configurations for acceptable performance
- Assess performance of current configurations
- Choose proper hardware capacity for a proposed configuration
- Choose the proper software programs for a proposed configuration
- Tune a new or current configuration

Throughout the report, ″**General Considerations**″ are identified. This information is specially identified as information to consider for tuning and capacity planning analysis. This does not imply that all information is summarized in these general considerations, because the information provided in this report can provide additional information dependent on the analyzer′s reference point.

This report focuses on the use of stored procedures for online transaction processing types of environments. Network delays play an important role in how well the environment performs. Through the use of stored procedures, network delay plays a lesser role, but is not eliminated. For example, each transaction consists of a stored procedure call and a commit (or rollback). This results in two ″round trip″ message flows from the client to the database. After the stored procedure completes, any locks created remain held until the commit or rollback flows across the wire. Without stored procedures, locks might be held much longer because there are two ″round trip″ messages for every SQL statement.

The burden of providing good throughput in a client/server environment falls on several different people:

- Network designers to make sure the network has capacity and an efficient path exists between the client and target database.
- Database administrators to make sure the target database is operating within capacity and tuned for client/server access.
- Application programmers to make sure the program is efficiently manipulating the database, for example, use of indexes.
- Any other administrators of software programs that might exist between the client and the target database (i.e. gateways).

The importance of this report is that it provides a reasonably tuned client/server database configuration. It achieved a certain level of throughput using a combination of components. Some questions to ask when analyzing this report to apply towards your environment are:

- How does the data throughput compare to my environment or proposed environment?
  - On the LAN?
  - With similar/different hardware?

© Copyright IBM Corp. 1996

- How much processor capacity would I need for my environment considering the throughput and transaction mix? DB2, etc...
- Is the response time at the clients reasonable, or can I trade worse response time for additional clients?
- How much network line capacity would I need considering throughput?
- How many network devices are between the clients and the database?

6. Other sources of information

- **DB2 Administration Guide For Common Server V2** S20H-4580
- **DRDA Connectivity Guide** SC26-4783