

## *Tuning your System with the Buffer Pool Tool® for DB2*

Joel Goldstein  
Thursday - August 17, 2006  
Session 1364

### **Abstract:**

**Effective pool tuning offers great opportunities for improving both system and application performance. Aside from the elapsed time and cpu savings from I/O elimination, this process often highlights significant application performance problems. This presentation will take you through the steps of analysis and tuning using the industry standard Buffer Pool Tool for DB2 from Responsive Systems, and show you how to analyze and tune your system, and evaluate the payback. Converting I/O savings to elapsed time, dollar savings, and improved productivity.**

**Aside from the elapsed time and cpu savings from I/O elimination, the tuning process often highlights significant application performance problems. See the steps of analysis and tuning using the industry standard Buffer Pool Tool for DB2, how to analyze and tune your system, and evaluate the payback. Converting I/O savings to elapsed time, \$ savings, and improved productivity.**

**Buffer Pool Tool is the only product that can predict the I/O rate per second. This is a measurable metric. Unlike Hit Ratios, it can be converted into CPU costs, and elapsed time savings for both online and batch functions.**

## Presentation Objectives

- Software Objectives
- Architecture
- Components and Functions
  - Mainframe and Workstation
    - Utilities
- Using Buffer Pool Tool
- Analyzing sets of data
  - Finding the problems and opportunities
- Illustrate before/after tuning performance
- Summaries and the next version – V8.3

**The objective of the presentation is to provide an understanding of the Buffer Pool Tool as a tuning product, its components, and capabilities.**

**The presentation will step through several sets of system data to illustrate the types of data and analysis that BPT provides, and to show how it predicts the effects of tuning changes.**

## Software Objectives

- Provide the ability to *predict the effect of buffer pool changes*
  - Simple changes – size & thresholds
  - Moving objects into other existing, or new, pools
  - *Predict the I/O rate/sec* – the only measurable metric
    - **Hit Ratios are interesting, but useless** as performance metrics
- Provide a reliable technique for grouping objects into multiple pools
  - Ramos and Samos ( and then working set size)
    - **Is the Industry Proven technique!!**
- Show the performance projections, and let you make the intelligent decisions based upon your resources

The Buffer Pool Tool® set the benchmark for DB2 buffer pool performance analysis and tuning approaches. The objective for the software, when it was designed, was to provide an easy to use, low overhead, reasonably priced product, that can predict the performance effect of changes to buffer pool configurations. It has been proven over the years that several to many pools are necessary to obtain good performance, and the proper grouping of objects into pools based upon access type and working set size yields the best performance. In most cases, using only two or three large pools does not provide the same level of performance and system throughput as multiple pools. The technique used to group objects into multiple pools is Ramos/Samos (random mostly, sequential mostly), and within these groups there are sub-groups based upon working set size. The wkset size is determined from object pool usage, and has no relationship to object statistics in the catalog.

## Software Objectives - 2

- Show you how to tune
- Show you the effect of changes
- Show you the rationale behind tuning changes
- Show you both system and application performance issues
- Learn how things work – your system & your applications
- Optimize system and application performance, and memory utilization
  - Memory isn't unlimited yet, and certainly not at most installations
    - Some large client systems are already ***past 6 Gigabytes*** of pool allocations

**Before the existence of the Buffer Pool Tool, pool tuning, beyond avoiding pool thresholds, was mostly a guessing game.**

**We had some basic guidelines, such as separating indexes from data - but there was nothing that could predict the effect of changes. This was not acceptable for important business systems.**

## **NOT - Software Objectives**

- Be a black box, and present only recommendations
  - You can see the reasons/rationale behind any tuning changes you make – *tuning should be a learning process*
- Change the system dynamically from snapshot statistics
  - All changes to be determined and implemented by the performance analyst
    - By the time active monitoring detects, and reacts, to a significant performance change, the system has already changed again...
    - **Too much overhead to do it properly.....**
    - **Not possible to predict from Statistics.... Averages of averages... you need a buffer manager and IO trace**
- Run/Monitor all the time
  - Continuous CPU consumption/overhead
  - Not necessary when tuning peak periods, or problem periods

**One of the essential parts of our approach is showing the analyst how all the objects are used, and the effect of changes at varying pool sizes. There are simple changes, such as increasing the pool size, and complex changes such as moving objects into different pools, and showing the performance impacts of the change at varying pool sizes.**

**We did not want to be a black box – and just present a change scenario. Tuning is a continual learning process – and it's not possible to factor all these scenarios into software, and keep it current. There is rarely a final solution, as every tuning scenario includes some trade-offs.**

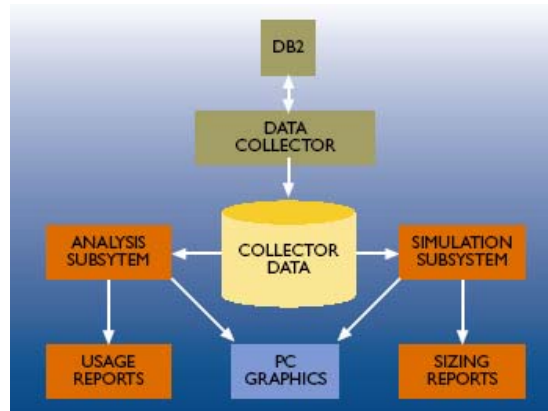
## Software Installation

- Receive emailed software as Tersed files
- Upload to the mainframe
- Un-Terse the files
- Edit the JCL
- Bind the BPT Plan to the DB2 Sub-System
- Ready to run....
  
- *About 20 Minutes of effort*

**KISS.....**

***Responsive Systems – Performance Software that Works!!***

**Overall System**



© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

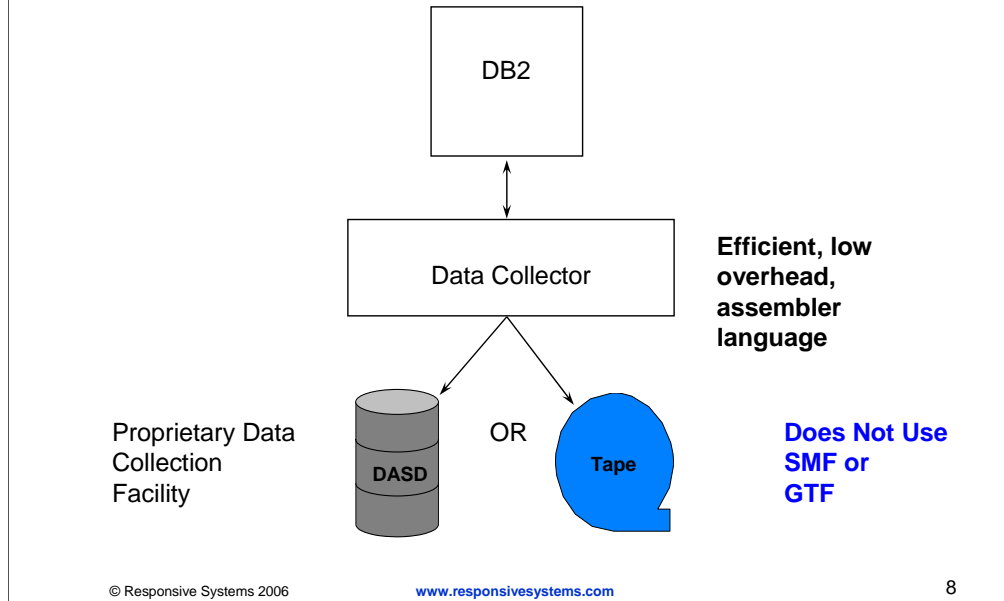
7

**The overall Buffer Pool Tool software system.**

**The individual components are discussed on the following slides.**

## Responsive Systems – Performance Software that Works!!

### Architecture - 1



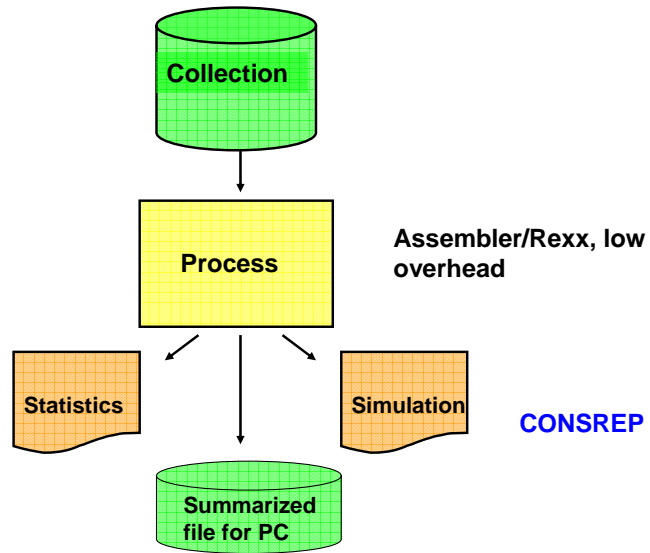
The Buffer Pool Tool has four components:

1. Collector that attaches to DB2, starts the buffer manager and I/O traces, and writes the data to our own proprietary dataset.
2. Statistical analysis that provides the un-paralleled level of pool and object information.
3. Simulation that predicts the performance of changes. Pool sizes, thresholds, and moving objects into different pools. Predicts hit ratios, wkset sizes, and *I/O rate/sec*.
4. Workstation component that provides graphic analysis, performance drill down, expert tuning, scan cost impacts, and clustering of wkset sizes for Ramos and Samos objects.

There are several utilities that provide object analysis, connection analysis, and page reference frequencies.

***Responsive Systems – Performance Software that Works!!***

**Architecture - 2**



© Responsive Systems 2006

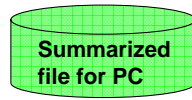
[www.responsivesystems.com](http://www.responsivesystems.com)

9

**Statistics and Simulation components provide detailed output reports, and small summarized files that are downloaded to the workstation for graphic analysis.**

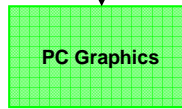
**Responsive Systems – Performance Software that Works!!**

**Architecture - 2**



Size - Few hundred K  
Stats, Sim

1-2 Meg Consolidated



Visual Basic



**System Summary**

Pool level detail & analysis

Expert tuning recommendations

Sequential scan costing

Dozens of analysis graphs

Cluster Analysis of Working Set Sizes

**PC files can be Statistics, Simulation, or Consrep that is a combined Statistics and Simulation file. The Consrep job produces the statistics and also performs a base simulation for all pools in use. This reduces the necessity to run initial simulations on all pools and then determine which additional simulations are necessary. You may not need to run simulations for every pool, only those that need tuning, based on high I/O rates.**

**Downloaded PC files are only a few hundred K in size, versus hundreds of Meg for a collection file.**

## Collecting Data

- What are your busiest or problem times?
  - This is when you want to collect data
    - At least twice, to determine workload consistency
- Depending upon system size/volume, collector should run up to one hour – really big systems, 5-15 Mins **(how big is BIG?)**
  - Full Buffer Manager and IO trace
    - Small snapshots, at intervals, lack statistical validity
      - National Bureau of Standards sampling techniques
  - Can collect up to 16 Gigabytes of data
- Low overhead process, 3-5% - only during the collection

**Buffer Manager and I/O traces are voluminous. You can not run this type of trace to SMF – you will lose massive amounts of DB2 and other vital system performance data. Using GTF is very high overhead, and GTF will wrap the dataset, invalidating all data. Using efficient assembler language, and discarding unnecessary data from each record keeps overhead at a minimum, and reduces the output dataset size. It's still very large. Several hundred Meg to 16 Gig, depending on system size and length of collection.**

## Collected Data... Now what 1 ?

- Run Statistics PCSTAT
  - Statistical analysis
    - Reports
    - PC file
- Run Simulation(s) PCSIM *Base Simulation*
  - Shows performance at varying pool sizes
    - Reports
    - PC file – *Wkset sizes*, as well as *I/O rates*
      - Base simulation – one size the same as the current size
- Consolidated Reporting **CONSREP**
  - *Does both of the above automatically*
  - Saves having to download multiple files to the workstation

**A Base Simulation starts at a slightly smaller pool size, and runs up substantially larger.**

**This shows performance at a smaller size, if the pool is oversized, and shows improvements if memory will reduce I/Os.**

**The prediction of Working Set Sizes (WKSET) is critical, since this is used to determine object groupings into pools.**

**The WKSET does not have any relationship to the number of object pages in the DB2 catalog.**

**Consolidated reporting automatically runs a simulation for each pool, and creates a combined file for PC download. This ranges from a few hundred K to a Meg in size. Compared to hundreds of meg or Gig for a collection file.**

## *Responsive Systems – Performance Software that Works!!*

### Collected Data... *Now what 2 ?*

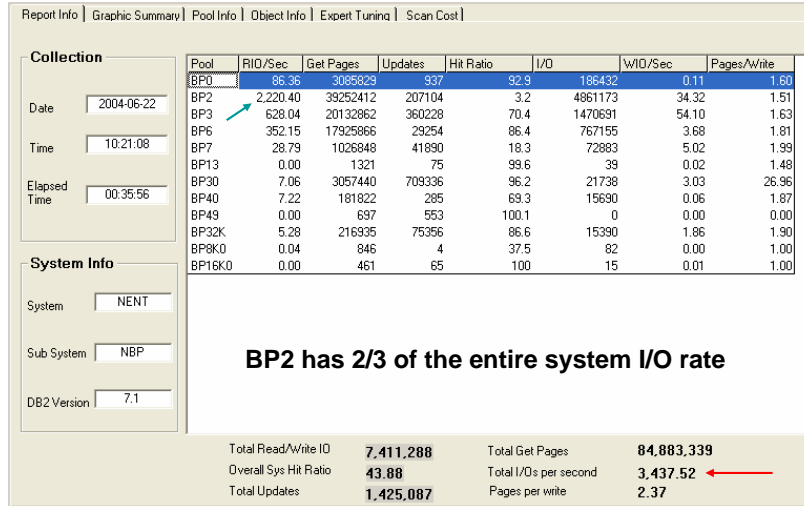
- In most cases we go to the workstation now
- We need the detail print reports in rare cases, since almost all the data is available on the workstation.
- We'll look at some data and analysis from several systems
  - *There is no absolute analysis path, it all depends on the performance data*



**EYEBALL technique - Some problem areas, or areas for potential savings will be obvious. Sometimes the solutions are not obvious at all. I see unusual and interesting things in client data several times a year, and they often require a bit of thought, digging around through data, to determine the cause and possible approaches to better performance.**

## Responsive Systems – Performance Software that Works!!

### Large System – Buffer Pools 160M GP/hr



© Responsive Systems 2006

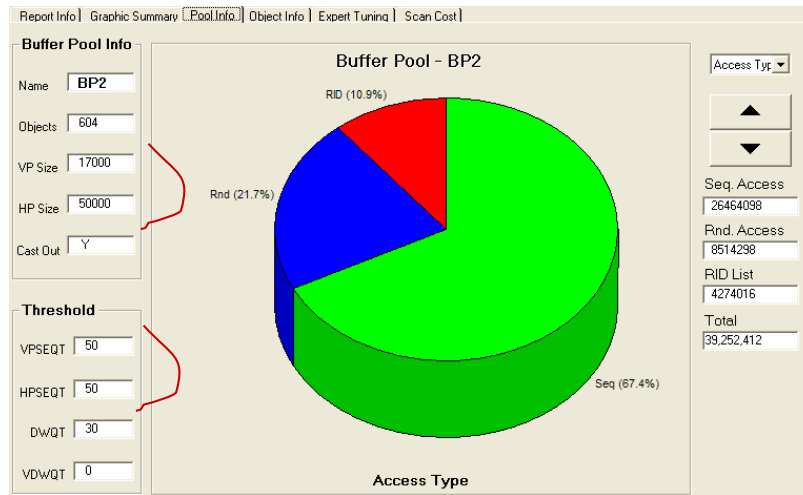
[www.responsivesystems.com](http://www.responsivesystems.com)

14

**This DB2 sub-system issues about 160 Million Getpages per hour.  
Not the largest I work with, but bigger than most in the industry...**

## BP2 – with the high I/O rate

Default graph at pool level



© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

15

**Large pool, mostly sequential access.**

## How much do you think SP Getpages cost?

Report Info | Graphic Summary | Pool Info | Object Info | Expert Tuning | Scan Cost

**Overall Collection Time** = 2,156.00  
**CPU Seconds Cost** = 2,381.77

System Scan Cost | Pool Scan Cost

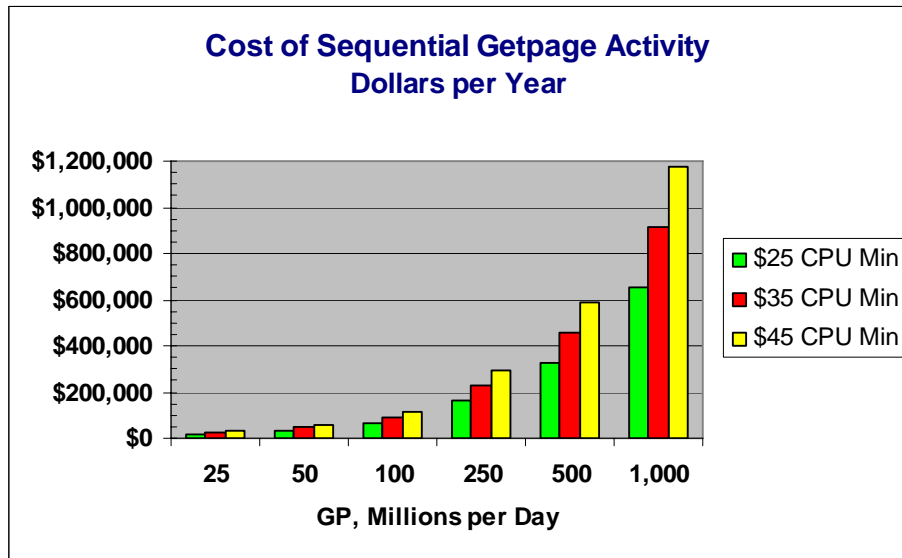
Object Name	Pool	Sequential GetPages	CPU-Seconds Cost
VBRK.VBRK	BP2	5226003	23.517
MARA.MARA	BP2	5192969	23.368
STAB10.SSTAB10	BP2	2625325	11.814
DSNDB07.DSN4K03	BP30	2186297	9.838
BTAB22.SBTAB22	BP2	1813674	8.162
KNA1.KNA1	BP2	1549904	6.975
STAB33.SSTAB33	BP2	1307923	5.886
LTAP.LTAP	BP2	1229362	5.532
BTAB19.SBTAB19	BP2	807689	3.635
LIKP.LIKP	BP2	680556	3.063
S501.S501	BP7	667841	3.005
DSNDB07.DSN4K04	BP30	586573	2.64
LFA1.LFA1	BP2	580853	2.614
STAB34.SSTAB34	BP2	552258	2.485
BKPF.BKPF	BP2	521264	2.346
AFRU.AFRU	BP2	514777	2.316
NAST.NASTX	BP2	443284	1.995
A14DXBSD.TPRIXPAX	BP2	418809	1.885
MAKT.MAKT	BP2	381471	1.717
VBUK.VBUK~Z1	BP3	373205	1.679

**35 Mins Elapsed**

© Responsive Systems 2006 [www.responsivesystems.com](http://www.responsivesystems.com) 16

**This is one DB2 subsystem, over a 35 minute period, used 2,381 seconds of CPU for sequential scan. Yes, this is more seconds than than 35 minutes – but this is what happens on a large machine with many engines, all working concurrently.**

**Responsive Systems – Performance Software that Works!!**



© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

17

**This is based on a 2064 processor with 210 MIP engine speeds.**

## Responsive Systems – Performance Software that Works!!

Same System – 7 Mos. Later 198M GP/hr

The screenshot shows the Buffer Pool Tool for DB2 - BP1 interface. It includes a navigation bar with tabs: Report Info, Graphic Summary, Pool Info, Object Info, Expert Tuning, Scan Cost, Sim Graph Analysis, and Sim Cluster Analysis. The main area is divided into several sections:

- Collection:** A table listing various buffer pools (BP0 to BP16K0) with columns for RI/O/Sec, Get Pages, Updates, Hit Ratio, I/O, WIO/Sec, Pages/Write, Write I/Os, and Pages Written.
- System Info:** Fields for System (NENT), Sub System (NBP), DB2 Version (7.1), and DS Group.
- Summary:** A section with red text stating: "Getpage rate is UP 20% added 2 more pools" and "IO rate is 1100 per second Lower".
- Summary Table:** A table with four columns: Metric, Value, Metric, Value. The values are: Total 4K Buffers 1,017,600; Total Read/Write IO 486,099; Total Get Pages 11,620,160; Overall Sys Hit Ratio 63.97; Total I/Os per second 2,348.30; Total Updates 204,512; Pages per write 2.55.

© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

18

**This DB2 sub-system now issues about 198 Million Getpages per hour.  
The I/O rate has dropped 1100 per second.**

## Responsive Systems – Performance Software that Works!!

### System Configuration – for a different system

Report Info | Graphic Summary | Pool Info | Object Info | Expert Tuning | Scan Cost |

**Collection**

Date: 2004-10-27  
Time: 09:41:34  
Elapsed Time: 00:06:44

Pool	RI0/Sec	Get Pages	Updates	Hit Ratio	I/O	wI0/Sec	Pages/Write
BP0	0.79	196402	110	99.6	349	0.08	3.13
BP17	0.82	510539	486458	99.5	332	0.00	0.00
BP20	491.67	4601921	167939	89.5	209591	27.12	3.50
BP21	363.74	7449826	282814	96.1	164129	42.52	2.82

**Performance Fallacy – just use a lot of memory, you don't need multiple pools.... Except, except, except ....**

Total Read/Write IO	<b>374,401</b>	Total Get Pages	<b>12,758,688</b>
Overall Sys Hit Ratio	<b>93.95</b>	Total I/Os per second	<b>926.74</b>
Total Updates	<b>937,321</b>	Pages per write	<b>3.08</b>

**System will generate 120 Million GP Hr.**

© Responsive Systems 2006 [www.responsivesystems.com](http://www.responsivesystems.com) 19

This is 6 minutes of data.

Over the course of an hour, this system will see 120M+ Getpages.

Four pools simply don't cut it....

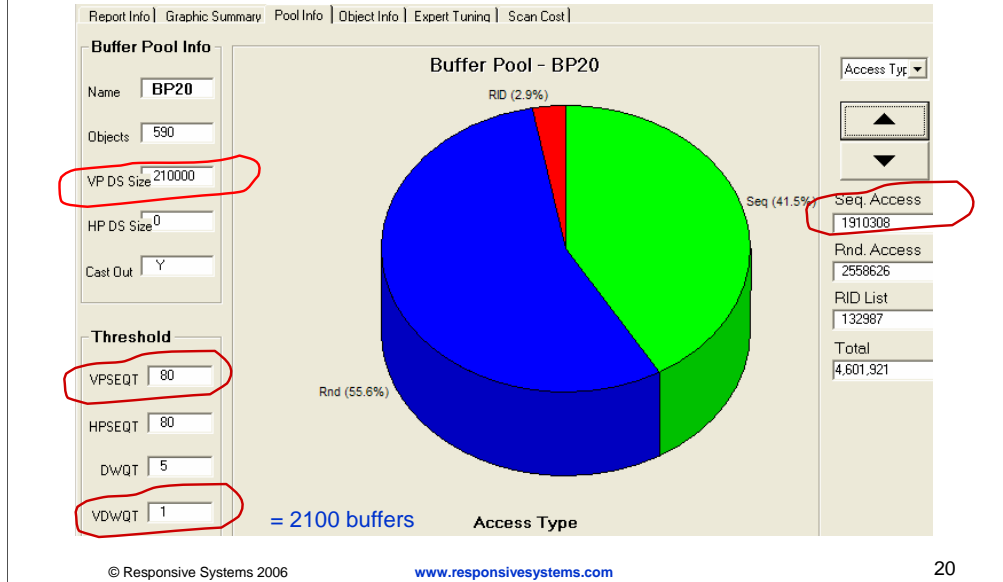
They used to have good separation of objects – until an external “consultant” told them they didn't need to do this, and they could get good performance with only two pools by throwing a lot of memory at them.

So their performance went to a very hot place, hand carried in a straw basket.

Aren't euphemisms great??

**Responsive Systems – Performance Software that Works!!**

**BP20 - 210,000 Buffers All the TS**



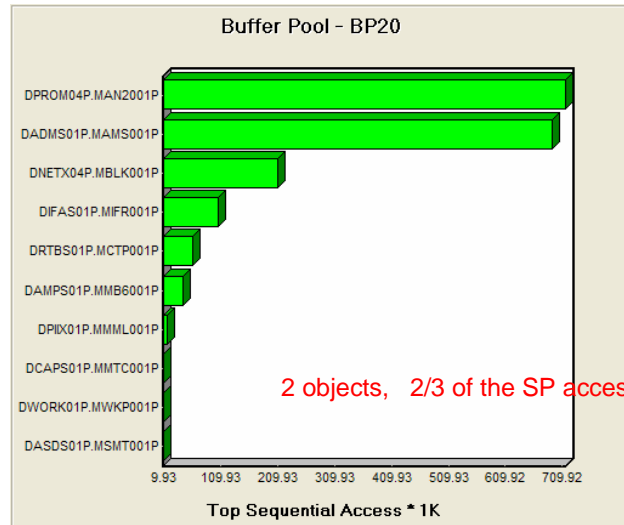
**This pool has all the Tablespaces, and 41.5% of the getpages are sequential.**

**So leaving the vpseqt at the default of 80% is a really good idea....**

***NOT!!***

**Responsive Systems – Performance Software that Works!!**

**BP20**



© Responsive Systems 2006

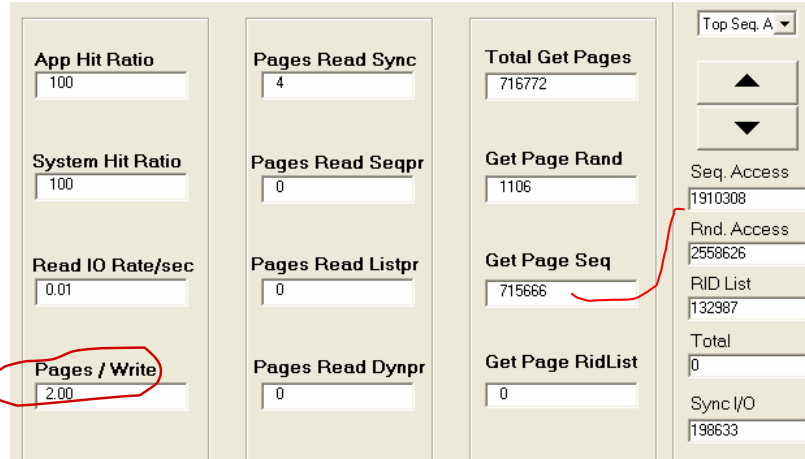
[www.responsivesystems.com](http://www.responsivesystems.com)

21

**Two heavy sequential objects are killing the pool performance.**

***Responsive Systems – Performance Software that Works!!***

**BP20 First heavy sequential**



**Memory Resident, 20% of pool GP, 1/3 of SP GP**

© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

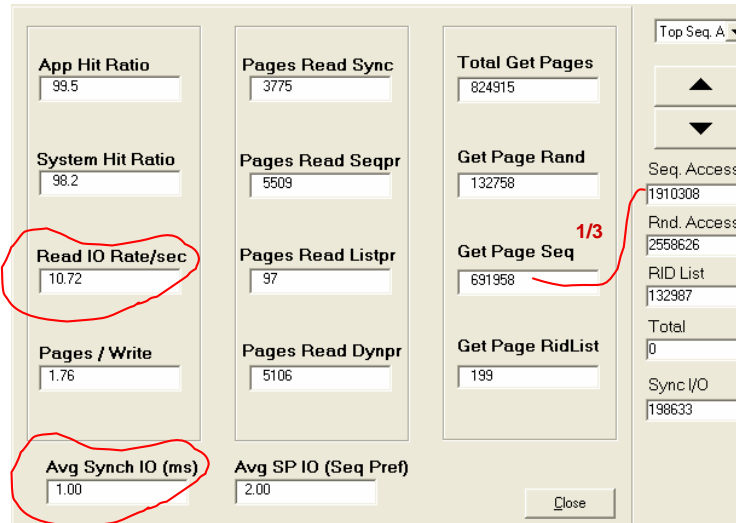
22

**Living in the pool.**

**More than 1/3 of the pool sequential access, and about 20% of the pool getpages.**

***Responsive Systems – Performance Software that Works!!***

**BP20 Second heavy sequential**



© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

23

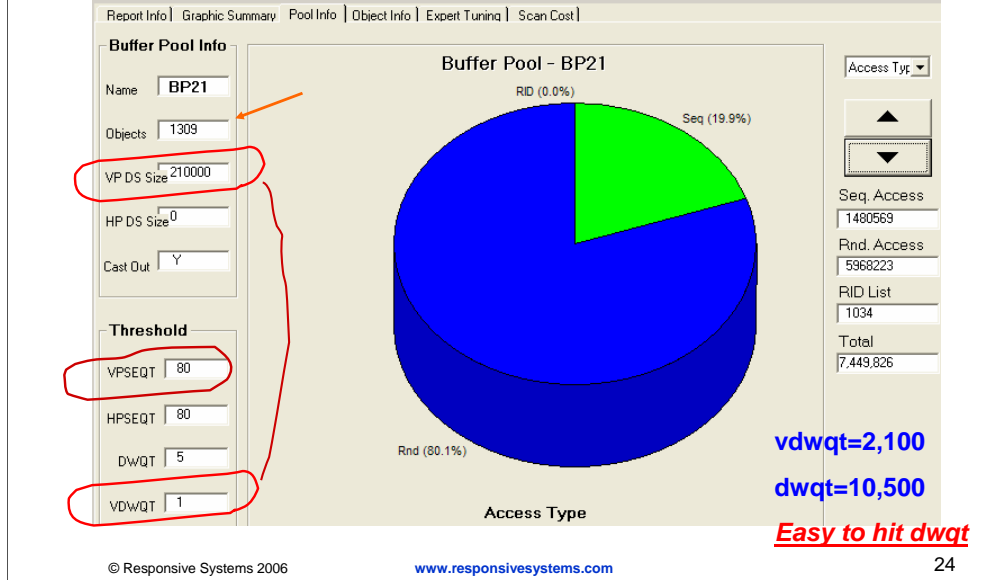
**Mostly in the pool.**

**Almost 1/3 of the pool sequential access.**

**Between these two objects we have 2/3 of the sequential access**

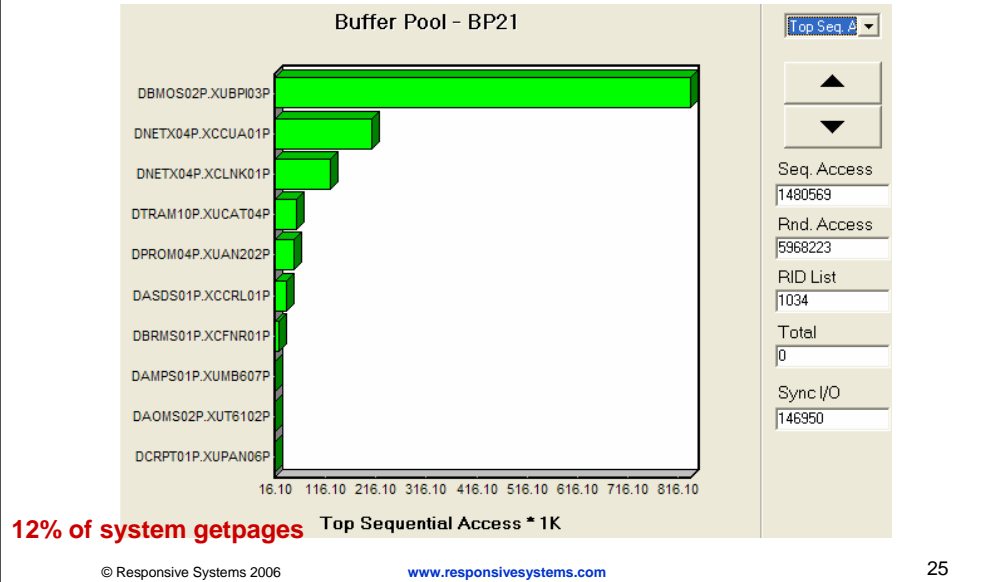
***Responsive Systems – Performance Software that Works!!***

**BP21 - all Indexes 210,000 Buffers**



**BP21 has all the Indexes....**

## BP21 Heavy sequential Index



**BP21 has all the Indexes....**

**Indexes normally shouldn't have much sequential access.**

**Of course there are some exceptions to this – when it's known, and planned that way.**

**Responsive Systems – Performance Software that Works!!**

**BP21 Heavy sequential Index - Resident**

<b>App Hit Ratio</b> 100	<b>Pages Read Sync</b> 11	<b>Total Get Pages</b> 843913	<b>Top Seq. A</b> ▲ ▼ Seq. Access 1480569 Pnd. Access 5968223 RID List 1034 Total 0 Sync I/O 146950
<b>System Hit Ratio</b> 100	<b>Pages Read Seqpr</b> 12	<b>Get Page Rand</b> 2922	
<b>Read IO Rate/sec</b> 0.03	<b>Pages Read Listpr</b> 0	<b>Get Page Seq</b> 840991	
<b>Pages / Write</b> 3.45	<b>Pages Read Dynpr</b> 0	<b>Get Page RidList</b> 0	
<b>Avg Synch IO (ms)</b> 3.00	<b>Avg SP IO (Seq Pref)</b> 28.00	<b>Close</b>	

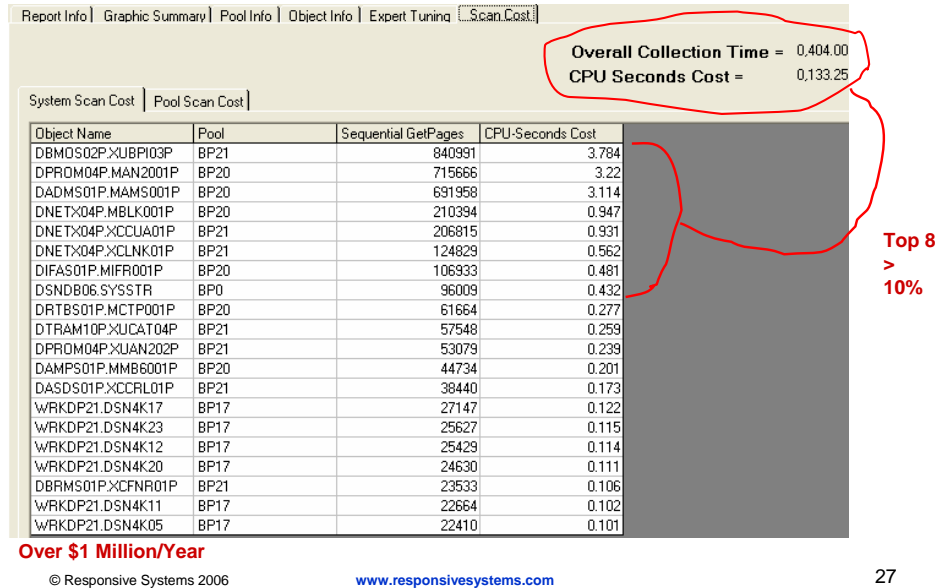
**60% of  
SP G/P**

**12% of system getpages, resident**

**This Index is memory resident, but the SP is eating CPU.**

## Responsive Systems – Performance Software that Works!!

### Sequential access costs.... \$1M per year



Within six ½ minutes of processing time, the sequential scanning of all objects cost 133 seconds of CPU.

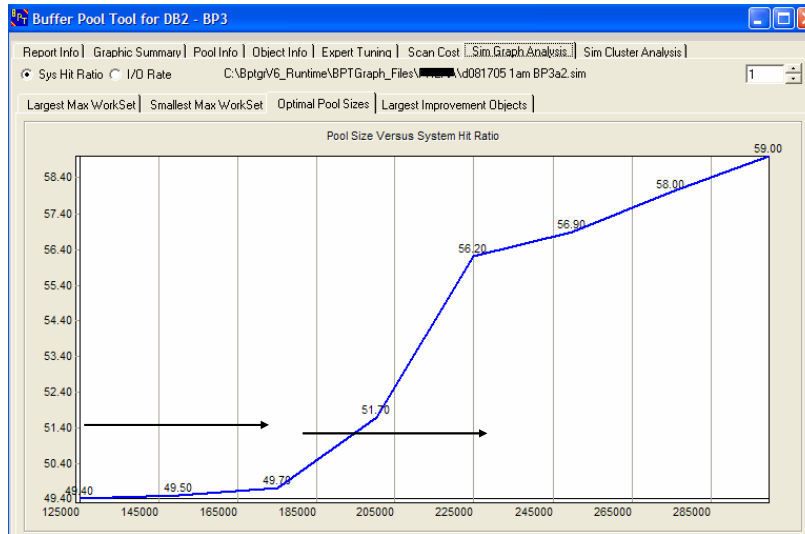
Based on a recent installation's internal chargeback costs of \$.46 a CPU second, this is \$61.

Multiplying it out, that's \$600 per hour, and > \$5000 per day, more than one million dollars per year.

And this is a company that's crying about their processor busy rate, and processing costs.

Costs are calculated by using the number of instructions for a getpage request, converting this to MIPS, and then CPU seconds based upon processor speed.

## What does a Hit Ratio really tell you?



**The first 50% does not get much payback**

© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

28

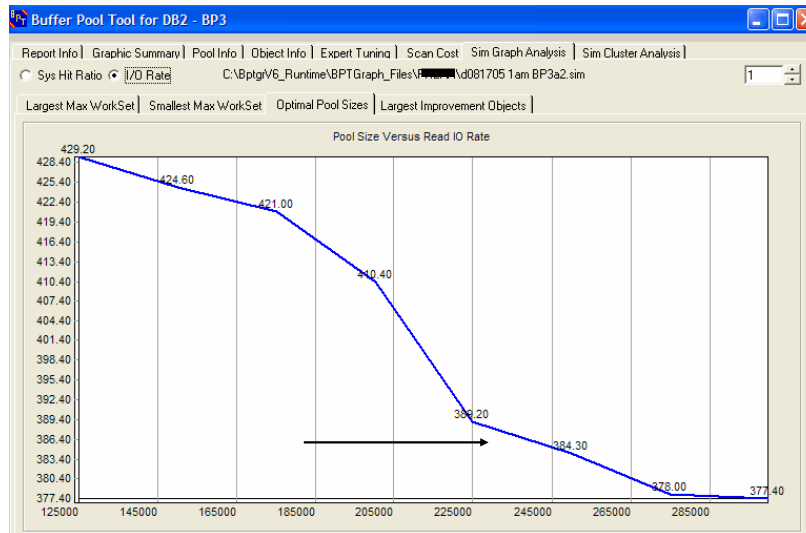
Ok, it shows you that performance is better. But how much better is it? How much CPU and elapsed times have been saved from I/O avoidance?

Increasing the pool by 50% does not give much payback, the next 50,000 shows a large improvement, and then the improvement curve flattens.

Again, it looks nice, but you can't take any of the numbers to the bank.

## ***Responsive Systems – Performance Software that Works!!***

### The I/O rate is a measurable Metric



**Why does the next 50% help so much?**

**A critical WKSET was reached**

© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

29

**The I/O rate is convertible into CPU costs, and elapsed time savings.**

**This is not just a suggestion to make the pool larger, it shows you the real benefit, and where to stop.**

**It shows you that the first 50,000 additional buffers don't provide much payback, but the next 50,000 give a huge payback.**

**The large payback from the second increment of 50,000 buffers is because we passed a critical working set threshold for a heavily accessed object. As stated earlier, the wkset size of an object has nothing to do with the number of pages shown in the catalog. It is the number of pages in the pool at a specific point in time.**

## Responsive Systems – Performance Software that Works!!

### What is a Working Set?

Buffer Pool Tool for DB2 - BP3

Report Info | Graphic Summary | Pool Info | Object Info | Expert Tuning | Scan Cost | Sim Graph Analysis | Sim Cluster Analysis

Pool Usage Intent:  Sequential  Random

Pool Size: 125000 Cluster Radius: 1.2

Object	Smallest Max	Largest Max
1	29735	37809
2	23121	25959
3	11038	19441
4	1391	9322
5	1	1376

Type	Object	Max Work Set
T	DDDLMO01.DBSEC03.TAY10DDL	29735
T	DLNSED01.DBSEC10.TLN80LNS	37809

**A WKSET is the number of pages in the pool at any point in time...**

**No relationship to DB2 catalog information**

**Track and calculate over time, not a momentary snapshot**

**Snapshots are useless and mis-leading**

© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

30

**Has no relationship to the number of pages reflected in the DB2 catalog. As an example, take an object with 1,000,000 pages.**

**The applications access 100,000 pages. The wkset will never be larger than 100,000 and will usually be quite less if it is in a pool with other objects. Wkset is also dependant upon overall pool activity, interactions with other objects, and reference of pages. Unfortunately, it's not a simple number we can easily determine without playing back all pool activity, and tracking the resident pages for all objects across time.**

## Responsive Systems – Performance Software that Works!!

As the pool size increases, the WKSET increases

The screenshot shows the Buffer Pool Tool for DB2 - BP3 interface. The 'Pool Usage Intent' is set to 'Random'. The 'Pool Size' is 275000 and the 'Cluster Radius' is 1.2. The 'Cluster Info' section contains two tables: 'Clusters' and 'Objects'.

Object	Smallest Ma	Largest Max
1	63781	80336
2	47687	57195
3	31277	42461
4	22512	27576
5	2625	19082
6	1	2487

Type	Object	Max Work S
T	DDDL001.DBSEC03.TAY10DDL	63781
T	DLNSED01.DBSEC10.TLN80LNS	80336

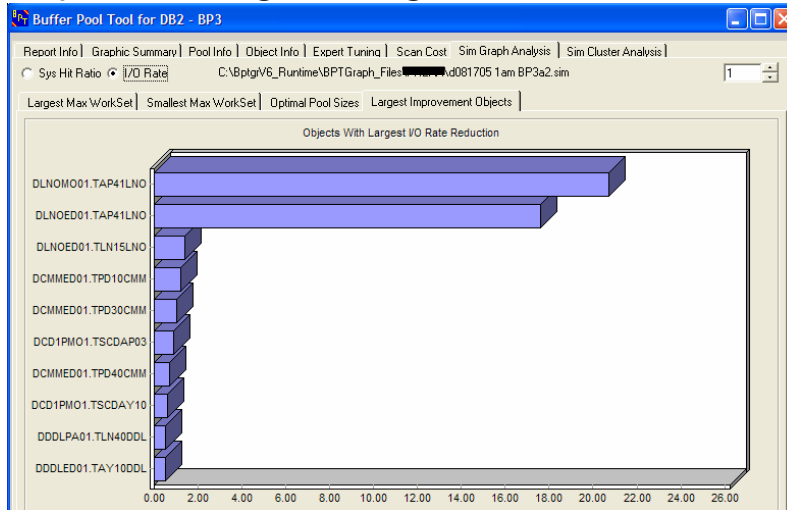
**Pool increased by 125% wksets increased by 114% and 112%**

**Other objects may have higher, or smaller, growth percentages**

**These use more pool resources, but don't get much I/O relief from the increased pool space.**

## ***Responsive Systems – Performance Software that Works!!***

### **Objects that got the greatest benefit...**



**Sometimes not as we expect...**

© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

32

**Looking at the objects that show the largest improvements can be surprising. Very often it isn't the objects you thought would get the most benefit.**

## Responsive Systems – Performance Software that Works!!

### An original Pool Size, Sequential Objects

Buffer Pool Tool for DB2 - BP3

Report Info | Graphic Summary | Pool Info | Object Info | Expert Tuning | Scan Cost | Sim Graph Analysis | Sim Cluster Analysis

Pool Usage Intent:  Sequential  Random

Pool Size: 125000 Cluster Radius: 2.8

Cluster Info

Object	Smallest Max	Largest Max
1	22131	40290
2	9856	16928
3	1703	8802
4	1	1440

Δ 5000

Type	Object	Max Work S
T	DLNOMD01.DBSEC06.TAP41LNO	40290
T	DCD1PM01.DBSEC14.TSCDAP03	25004
T	DLNOED01.DBSEC06.TAP41LNO	25076
T	DCD1PM01.DBSEC14.TSCDAY10	22131

© Responsive Systems 2006 [www.responsivesystems.com](http://www.responsivesystems.com) 33

We should always look at the wkset size of the SP objects in a pool, to see their impact on the random objects.

## Larger Pool Size

The screenshot shows the 'Buffer Pool Tool for DB2 - BP3' application. The 'Pool Usage Intent' is set to 'Sequential'. The 'Pool Size' is 275000 and the 'Cluster Radius' is 2.8. The 'Cluster Info' section contains two tables:

Clusters		
Object	Smallest Max	Largest Max
1	39522	53216
2	16886	24128
3	2528	10816
4	1	1824

A red bracket highlights the difference between the largest and smallest maximum values in the clusters table, labeled as  $\Delta 15,000$ .

Objects		
Type	Object	Max Work S
T	DLNOMD01.DBSEC06.TAP41LNO	40290
T	DCD1PMD1.DBSEC14.TSCDAP03	53216
T	DLNOED01.DBSEC06.TAP41LNO	39522

Below the tables, the text reads: "Also one fewer object in cluster".


© Responsive Systems 2006 [www.responsivesystems.com](http://www.responsivesystems.com) 34

**How much do they grow if add memory to the pool?**

## Responsive Systems – Performance Software that Works!!

### Max Wkset does not grow – 1<sup>st</sup> Object

Pool Size	125000	150000	175000	200000	225000	250000	275000	300000
Get Pages	1605091	1577664	1553950	1529814	1518617	1514142	1507511	1489735
Avg Work Set	4638	5462	6281	7105	7544	8008	8455	8773
Max Work Set	40290	40290	40290	40290	40290	40290	40290	40290
Sytem Hit Rat...	1.3	1.2	1.3	1	57.2	59.1	63.7	67.8
I/O per second	30.5	30.2	29.8	29.4	13.1	12.5	11	9.8

  
**Payoff**

**But Avg Wkset does grow - note the  $\Delta$  as the pool size increases**

© Responsive Systems 2006 [www.responsivesystems.com](http://www.responsivesystems.com) 35

**Note the payoff point, where the I/O rate takes a large drop....**


## Responsive Systems – Performance Software that Works!!

### Wkset does grow, initially.. – 2nd Object

Report Info | Graphic Summary | Pool Info | Object Info | Expert Tuning | Scan Cost | Sim Graph Analysis | Sim Cluster Analysis |

Sys Hit Ratio  I/O Rate C:\Bptg\V6\_Runtime\BPTGraph\_Files\BPTG\Ad081705 1am BP3a2.sim

Pool Size	125000	150000	175000	200000	225000	250000	275000	300000
Get Pages	1631778	1631778	1631778	1631778	1631778	1631778	1631778	1631778
Avg Work Set	4296	5205	6120	7028	7677	8268	8826	9406
Max Work Set	25076	30060	35056	39522	39522	39522	39522	39522
Sytem Hit Rat...	0.3	0.4	0.4	25.9	35	41.6	48.6	57.2
I/O per second	33.3	33.4	33.5	26.2	23.4	21.1	18.6	15.9

  
**Payoff**

© Responsive Systems 2006 [www.responsivesystems.com](http://www.responsivesystems.com) 36

**Note the payoff point, where the I/O rate takes a large drop....**

## Different DB2 System

Report Info | Graphic Summary | Pool Info | Object Info | Expert Tuning | Scan Cost | Sim Graph Analysis | Sim Cluster Analysis


**Collection**

Date: 2006-02-21  
Time: 08:07:05  
Elapsed Time: 00:12:04

Pool	RIO/Sec	Get Pages	Updates	Hit Ratio	I/O	WIO/Sec	Pages/Write	Write I/Os	Pages Written	Avg Pg Res	Rand Pg
BP0	0.40	374004	1038	99.9	378	0.12	4.34	85	369	722	
BP1	0.08	73107	21238	99.9	7644	10.48	1.40	7589	10644	723	
<b>BP3</b>	<b>793.68</b>	<b>4446068</b>	<b>106929</b>	<b>73.6</b>	<b>585419</b>	<b>15.01</b>	<b>1.90</b>	<b>10866</b>	<b>20630</b>	<b>532</b>	<b>454</b>
BP4	511.13	6475189	198617	89.7	406347	50.13	1.63	36292	59317	649	354
BP7	1.22	400574	282402	99.7	1226	0.47	22.25	340	7564	721	
BP32K	0.69	1250	166	59.3	519	0.02	2.25	16	36	429	

**System Info**

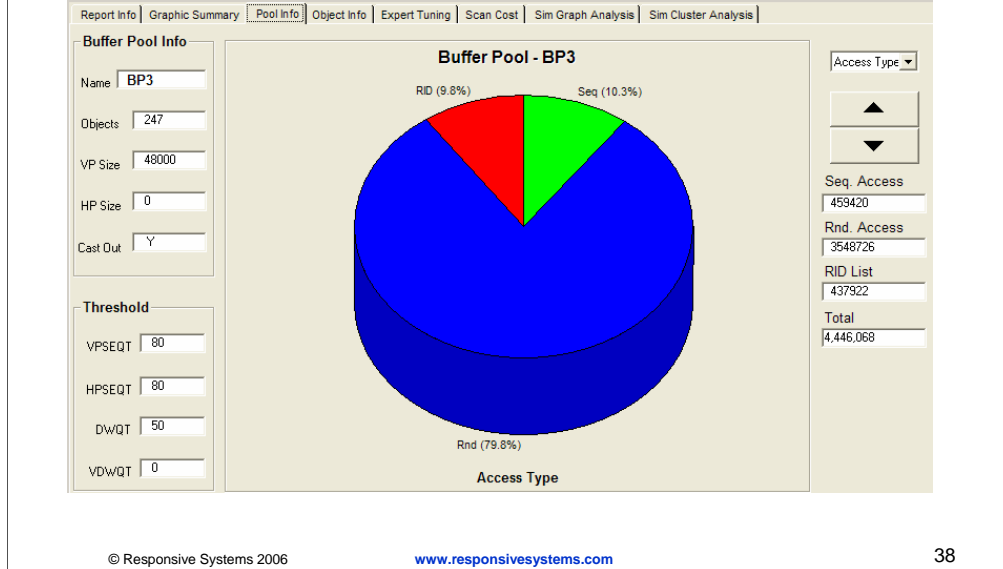
System: SYSA  
Sub System: DB2P  
DB2 Version: 7.1  
DS Group: DB2P



Total Read/Write IO: **1,001,533**      Total Get Pages: **11,770,192**  
Overall Sys Hit Ratio: **84.35**      Total I/Os per second: **1,383.33**  
Total Updates: **610,390**      Pages per write: **1.79**

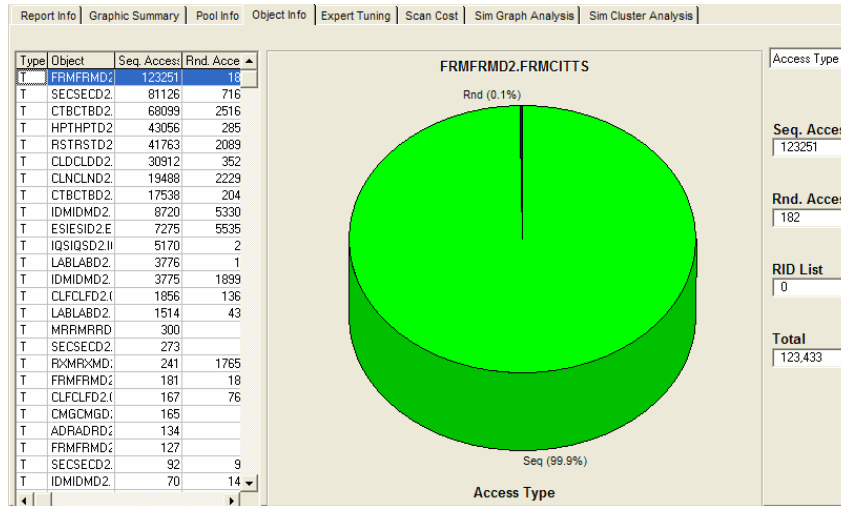
**Overall summary, few pools, two heavy I/O pools. BP3 is 57% of the system I/O**

## Different System



**BP3 with the heaviest I/O. 10% of the access is SP, so lets see which objects are accessed that way.**

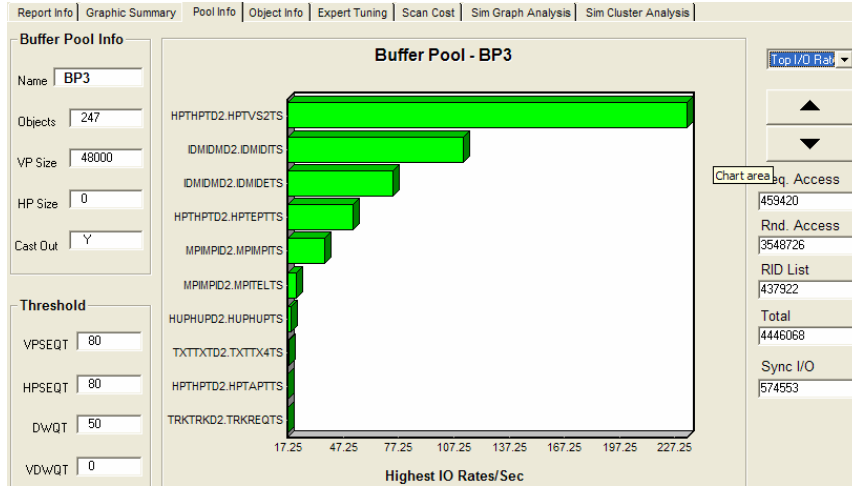
## The objects in BP3, sorted by SP access



**It's easy to see all the heavy hitters that are impacting the random objects.**

**Correcting some of the application access, will save lot's of CPU.**

## Largest contributors to the pool I/O rate



**One really heavy hitter...**

## Metrics for the heaviest I/O object

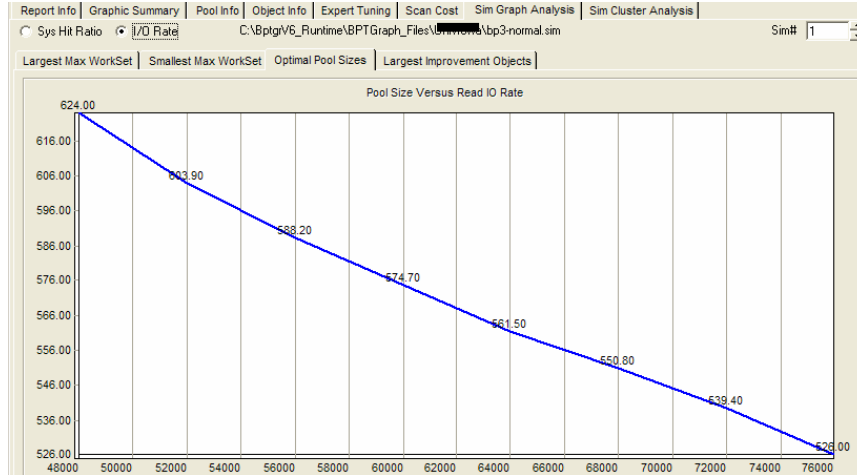
Report Info   Graphic Summary   Pool Info   Object Info   Expert Tuning   Scan Cost   Sim Graph Analysis   Sim Cluster Analysis			
<b>Buffer Pool Info</b>			
Name	BP3		
Objects	247		
VP Size	48000		
HP Size	0		
Cast Out	Y		
<b>Threshold</b>			
VPSEQT	80		
HPSEQT	80		
DWQQT	50		
VDWQQT	0		
<b>App Hit Ratio</b>	94.7	<b>Pages Read Sync</b>	82189
<b>System Hit Ratio</b>	81.2	<b>Pages Read Seqpr</b>	0
<b>Read IO Rate/sec</b>	233.33	<b>Pages Read Listpr</b>	182939
<b>Pages / Write</b>	1.51	<b>Pages Read Dynpr</b>	27280
<b>Avg Synchron IO (ms)</b>	3.00	<b>Avg SP IO (Seq Pref)</b>	0.00
<b>Total Get Pages</b>	1555311	<b>Get Page Rand</b>	1194957
<b>Get Page Seq</b>	0	<b>Get Page Seq</b>	0
<b>Get Page RidList</b>	360354		
Close			

**30% of the BP3 I/O rate**

**233 is a very high I/O rate for one object. High Synchron I/Os, but lot's of LP pages read too.**

## Responsive Systems – Performance Software that Works!!

### Base simulation



**30,000 buffers saves 100 I/O Sec**

© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

42

**Effect of giving BP3 more memory – 30,000 buffers will save about 100 I/O Sec.**

## Cluster analysis of the Wksets

The screenshot shows the 'Cluster Info' section of the software. It includes a 'Pool Usage Intent' section with 'Sequential' and 'Random' radio buttons, and a 'Pool Size' dropdown set to 68000. Below are two tables: 'Clusters' and 'Objects'.

Object	Smallest Max	Largest Max
1	20316	26408
2	4020	8579
3	668	3670
4	1	545

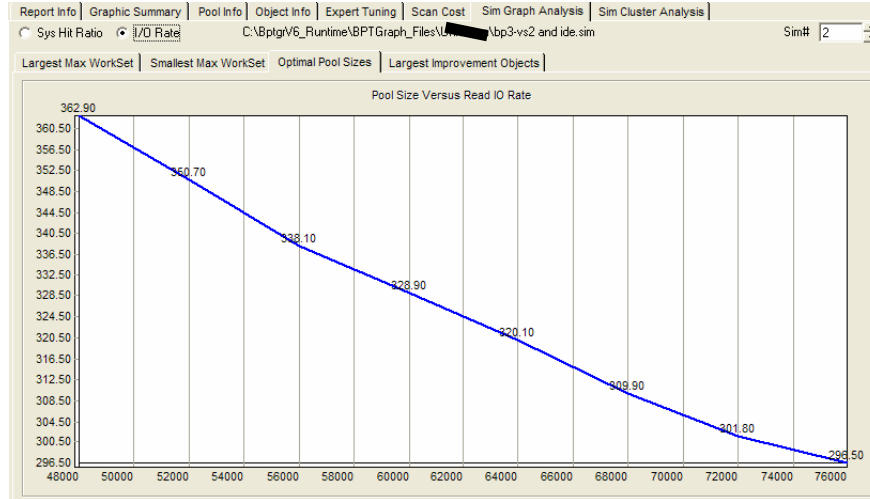
Type	Object	Max Work S
T	HPTHPTD2.DBA.HPTV52TS	20916
T	IDMIDMD2.DBA.IDMIDETS	26408

Two very large objects, may not belong in this pool

**Two very large objects, the vast majority of the objects in the pool are much smaller.**

**Perhaps they should be moved out of this pool...**

## Exclude – effect of taking them out of the pool



**Saves 260 I/O per Sec.**

© Responsive Systems 2006

[www.respondivesystems.com](http://www.respondivesystems.com)

44

**Taking out two objects saves 260 I/Os per second.**

## Include into new pool

Component: >>> DB2 Buffer Pool Simulation <<<

Results of Simulation for Buffer Pool.....BP5

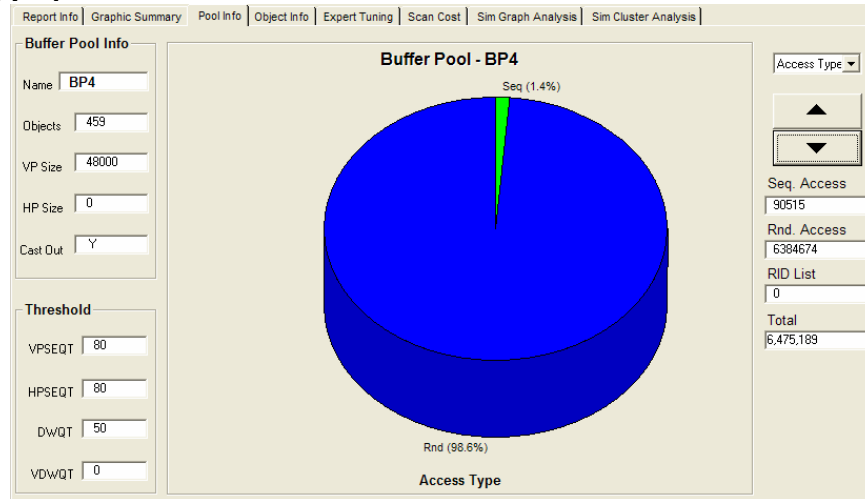
Bpool GetP total.....1,639,783

Bpool Size	GetP used	Num. of Hits	ApHit Ratio	Elapsed Time
12,000	1,611,264	1,465,361	91.0 %	00:11:47
16,000	1,606,878	1,471,613	91.6 %	00:11:44
20,000	1,601,930	1,474,626	92.1 %	00:11:41
24,000	1,593,240	1,473,702	92.5 %	00:11:38
28,000	1,561,755	1,448,138	92.8 %	00:11:32
32,000	1,549,369	1,441,324	93.1 %	00:11:25

Bpool Size	Pages Read	Read I/O	SyHit Ratio	Norm. I/O Rate
12,000	370.9 /S	212.4 /S	83.7 %	9.3 %
16,000	341.7 /S	197.8 /S	85.0 %	8.7 %
20,000	322.2 /S	187.1 /S	85.9 %	8.2 %
24,000	305.1 /S	176.6 /S	86.6 %	7.7 %
28,000	294.0 /S	169.6 /S	87.0 %	7.5 %
32,000	284.1 /S	163.0 /S	87.4 %	7.2 %

**We can put them into a new pool, and have better performance than just making the pool larger.**

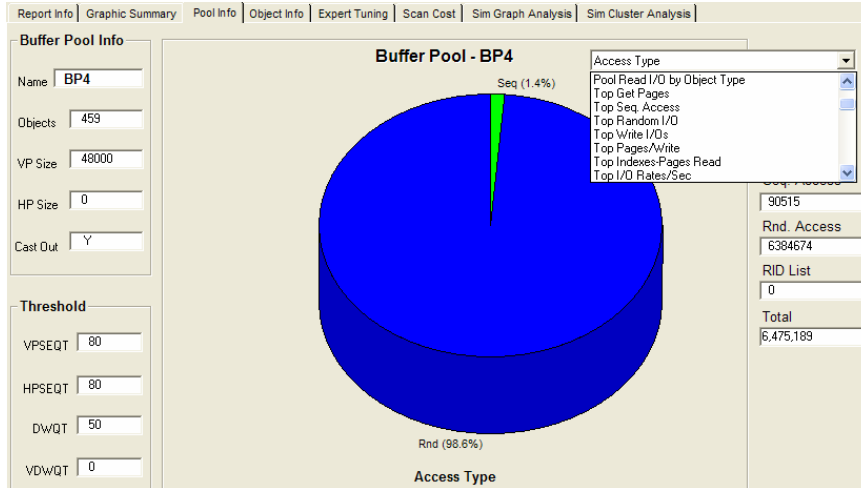
## BP4



**Almost all random access for BP4**

**Responsive Systems – Performance Software that Works!!**

**BP4**



Two dozen analysis graphs....

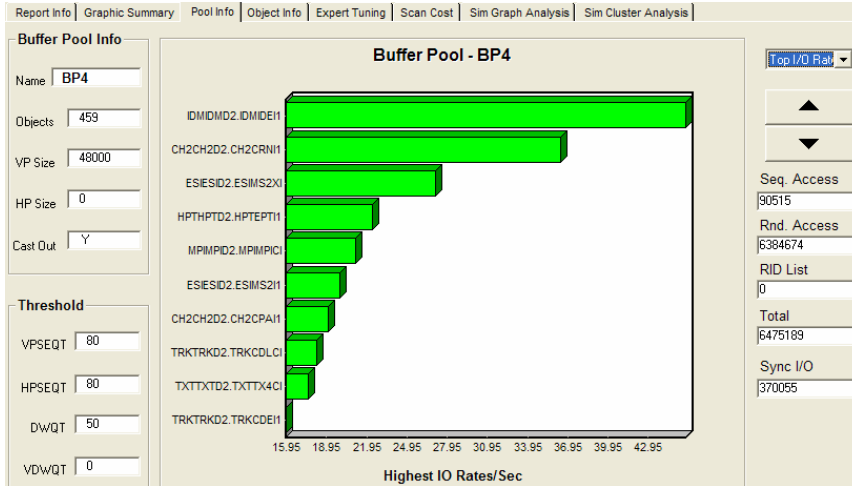
© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

47

**There are many graph options to help the analysis process.**

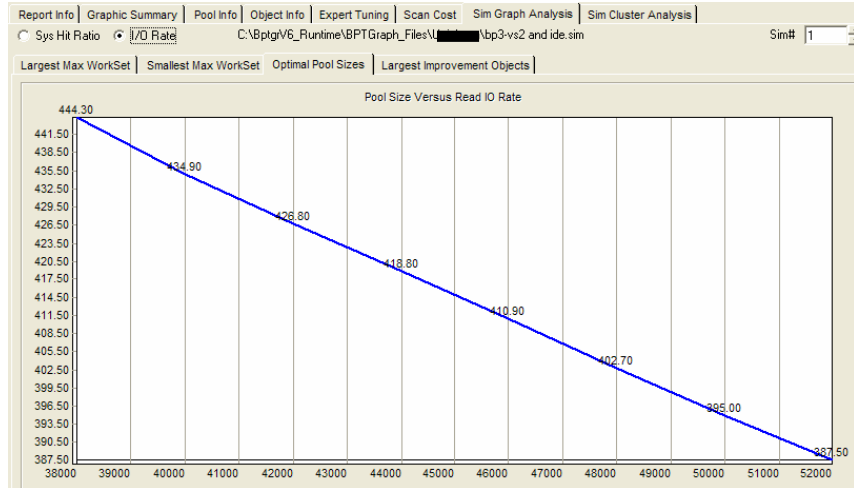
## BP4



**The heaviest I/O objects in BP4**

## ***Responsive Systems – Performance Software that Works!!***

### **BP4**



© Responsive Systems 2006

[www.respondivesystems.com](http://www.respondivesystems.com)

49

**Overall summary, few pools, two heavy I/O pools**

## Responsive Systems – Performance Software that Works!!

One large

Report Info | Graphic Summary | Pool Info | Object Info | Expert Tuning | Scan Cost | Sim Graph Analysis | Sim Cluster Analysis

Pool Usage Intent  
 Sequential  Random C:\Bptgr\6\_Runtime\BPTGraph\_Files\... pp3-vs2 and ide.sim Sim # 1 Pool Size 48000

Cluster Info

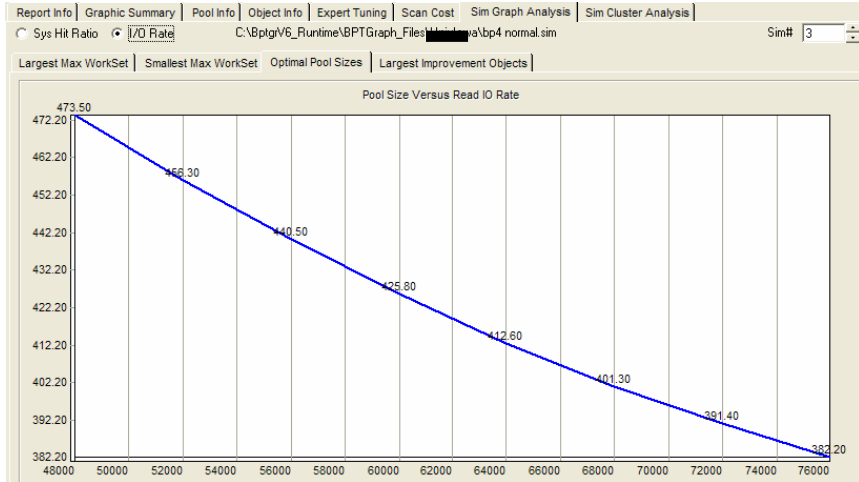
Object	Smallest Max	Largest Max
1	15636	15636
2	8750	9045
3	5213	5213
4	3227	4205
5	1827	2448
6	245	1708
7	1	239

Type	Object	Max Work S
I	IDMIDMD2.DBA.IDMIDI4	15636

One large object – perhaps this should be moved out...

**One very large random object. Perhaps we should simulate moving it out of this pool.**

## Make the pool larger



© Responsive Systems 2006

[www.respondivesystems.com](http://www.respondivesystems.com)

51

**How many I/Os can we save by making the pool larger?**

## Responsive Systems – Performance Software that Works!!

### Put it into a new pool

Results of Simulation for Buffer Pool.....BP6  
Bpool GetP total.....162,392

Moving to a new  
pool provides  
better overall  
performance

Bpool Size	GetP used	Num. of Hits	ApHit Ratio	Elapsed Time
12,000	139,769	123,354	88.3 %	00:10:19
14,000	137,578	122,590	89.2 %	00:10:16
16,000	134,934	121,677	90.2 %	00:10:11
18,000	134,082	121,914	91.0 %	00:10:09
20,000	132,577	121,396	91.6 %	00:10:06
22,000	130,489	120,368	92.3 %	00:10:02

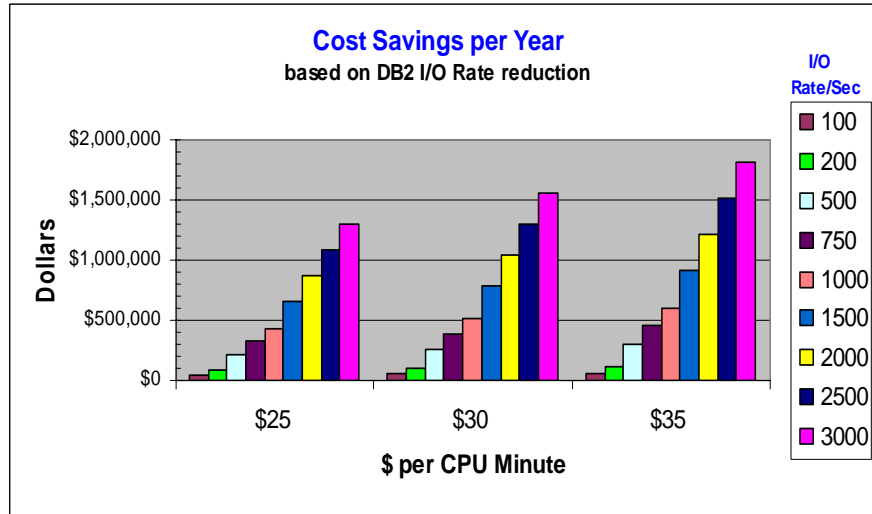
  

Bpool Size	Pages Read	Read I/O	SyHit Ratio	Norm. I/O Rate
12,000	94.8 /s	30.6 /s	58.0 %	13.6 %
14,000	82.0 /s	28.0 /s	63.3 %	12.5 %
16,000	70.9 /s	25.0 /s	67.9 %	11.3 %
18,000	62.0 /s	22.8 /s	71.9 %	10.4 %
20,000	54.5 /s	21.0 /s	75.1 %	9.6 %
22,000	44.3 /s	19.2 /s	79.6 %	8.9 %

**Moving the object into a new pool, gives better performance than just making the original pool larger.**

***Responsive Systems – Performance Software that Works!!***

## Eliminating I/O Saves Money !!



© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

53

**These I/O rate per sec savings, up through 1,000 per second, have been achieved by clients using Buffer Pool Tool.**

## Responsive Systems – Performance Software that Works!!

### Last large System – Tuning Now....200M GP/Hr

The screenshot shows the Buffer Pool Tool for DB2 - BPO interface. It includes a menu bar with options like Report Info, Graphic Summary, Pool Info, Object Info, Expert Tuning, Scan Cost, Sim Graph Analysis, and Sim Cluster Analysis. On the left, there are input fields for Date (2006-02-27), Time (12:01:26), and Elapsed Time (00:03:00). Below these are System Info fields for System (SYZ2), Sub System (DPW2), DB2 Version (8.1), and DS Group (DPW/G). The main area contains a table with 12 columns: Pool, R/I/O/Sec, Get Pages, Updates, Hit Ratio, I/O, W/I/O/Sec, Pages/Write, Write I/Os, Pages Written, and a small icon. The table lists 13 buffer pools (BP0 to BP12) with their respective statistics. At the bottom, there is a summary section with the following data:

Pool	R/I/O/Sec	Get Pages	Updates	Hit Ratio	I/O	W/I/O/Sec	Pages/Write	Write I/Os	Pages Written	Icon
BP0	173.67	311326	42	25.3	31263	0.01	1.50	2	3	
BP1	1,797.01	3618533	72231	73.4	326365	16.13	5.73	2904	16636	
BP2	1,741.15	1207515	113379	27.1	315393	11.03	14.91	1986	29603	
BP3	5.82	495508	357766	99.4	1047	0.00	0.00	0	0	
BP4	354.53	130087	236	25.6	63836	0.11	7.05	20	141	
BP5	255.49	1997672	0	91.3	45989	0.00	0.00	0	0	
BP6	468.21	506337	216	76.5	84299	0.12	5.57	21	117	
BP7	9.91	19927	35	51.6	1791	0.04	15.13	8	121	
BP8	320.41	149384	49649	56.7	59484	10.06	18.40	1811	33319	
BP9	555.14	578578	18	81.3	99931	0.03	1.83	6	11	
BP10	389.24	144112	27	41.3	70064	0.01	3.00	1	3	
BP11	42.86	546021	25481	94.6	8179	2.58	10.93	465	5082	
BP12	150.82	357332	0	75.8	27147	0.00	0.00	0	0	

Summary statistics at the bottom of the interface:

Total 4K Buffers	546,000	Total Read/Write IO	1,134,788	Total Get Pages	10,062,332
Overall Sys Hit Ratio	71.66	Total I/Os per second	6,304.38	Pages per write	11.77
Total Updates	619,080				

© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

54

**This system is over 200Million Getpages/Hour, with a very high I/O rate.**

## Responsive Systems – Performance Software that Works!!

### Last large System – after Tuning .. 223M GP/Hr

Pool	RIO/Sec	Got Pages	Updates	Hit Ratio	I/O	WIO/Sec	Pages/Write	Write I/Os	Pages Written	Avg
BP0	7.47	62196	109	79.4	1906	0.00	0.00	0	0	
BP1	1,065.63	5841602	145332	85.3	276834	20.38	10.20	5198	53023	
BP2	669.47	1746550	162273	57.4	224564	11.10	16.49	2050	47003	
BP3	7.19	691664	481468	99.2	1833	0.00	0.00	0	0	
BP5	9.23	2540312	0	99.2	2353	0.00	0.00	0	0	
BP6	298.95	925587	291	90.6	76255	0.09	4.43	23	102	
BP7	3.95	10731	97	94.1	1043	0.14	21.03	35	764	
BP8	145.64	123651	25702	67.3	45712	33.62	25.09	8574	215079	
BP9	190.70	843507	24	94	50693	0.01	1.67	3	5	
BP10	145.45	99983	24	62.9	37091	0.00	2.00	1	2	
BP11	111.32	536843	11167	86.5	29302	3.59	18.51	915	16941	
BP12	7.85	754492	0	99.2	2001	0.00	0.00	0	0	
BP13	81.88	1289201	308	91.8	20891	0.04	13.91	11	153	
BP14	275.82	339407	1499	54.7	70427	0.36	14.59	93	1357	

Total 4K Buffers **744,000**

Total Read/Write IO **841,006**      Total Get Pages **15,813,806**  
Overall Sys Hit Ratio **86.13**      Total I/Os per second **3,298.06** ←  
Total Updates **828,294**      Pages per write **18.89**

**GP rate is up 10%, IO rate decreased 3,000/Sec**

© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

55

This system is over 223Million Getpages/Hour, with a greatly reduced, but still high I/O rate.

**Buffer Pool Tool V8.3**

- ***New Features and Capabilities***
  - Prediction of Group Buffer Pool performance
    - GBP performance itself, at varying sizes, and how the varying sizes impact the local pools on all the members
    - Reports – actual object access and usage across all the members of the group
    - Processes the concurrent activity for Groups up to 32 members
    - Cross system replication factor – simultaneous cache frequency of pages in local member pools
  - Up to a **50% reduction** of both **elapsed times and CPU** for **Statistics and Simulations**

**Buffer Pool Tool V8.3**

- ***New Features and Capabilities - Continued***
  - Partition Performance Analysis
    - Statistical analysis at both Object Summary and Partitions
    - Simulation/Prediction of I/O performance at the Object *individual Partition level*
  - Perform up to 16 different pool size simulations in one pass

## ***Responsive Systems – Performance Software that Works!!***

### BPTV8.3 Group Buffer Pool Data

The screenshot shows the 'Buffer Pool Tool for DB2 - BP1' application window. The 'GBP Data' tab is selected, displaying a grid of configuration and performance parameters for a Group Buffer Pool. The parameters are organized into four columns:

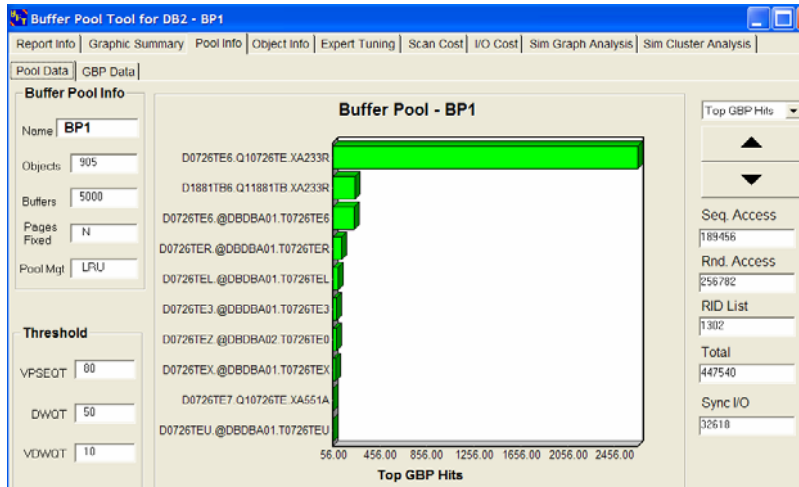
Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
CLASST	10	Duplex	D	Data Entries	4133	Data Entries Secondary	4133
POOLT	30	Check Point	5	Directory Entries	20669	Directory Entries Secondary	20669
Auto REC	Y	Directory Data Ratio	5.0	4K Pages	6144	4K Pages Secondary	6144
Cache	Y	GBP Hits	5470	% Getpages Hit	0.01		

Adding the GBP hit ratio and other data....

**Additional data about GBP performance has been added.**

***Responsive Systems – Performance Software that Works!!***

**BPTV8.3 Group Buffer Pool Object Performance**



Shows the Top 10 Objects that had pages retrieved from the GBP

**The bar for any object can be double-clicked to drill down to the object performance data.**

## ***Responsive Systems – Performance Software that Works!!***

### BPTV8.3 adds I/O Cost Analysis Tab

**Collection**

Pool	RIQ/Sec	Get Pages	Updates	Hit Ratio	I/O	WIO/Sec	Pages/Write	Write I/Os	Pages Written	Av
BP0	18.89	385140	12664	92	5102	0.59	4.77	155	739	
BP1	124.51	447640	151066	31.2	33672	4.02	4.61	1054	4651	
BP9	0.10	6503	6474	97.6	27	0.00	0.00	0	0	

Date: 2006-02-17  
Time: 15:32:25  
Elapsed Time: 00:04:22

**Overall Collection Time =** 262.000      **System CPU Seconds Cost =** 1.2405  
**Projected System 24Hr IO CPU Cost =** 409.09      **BP1 CPU Seconds Cost =** 1.076  
**Projected System 24 Hrs IO Delay =** 5,085.40      **System IO Delay Secs =** 15.42

Object Name	Pool	Total I/O	CPU Sec Cost	IO Elap Sec	24Hrs IO Delay
...	...	...	...	...	...

© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

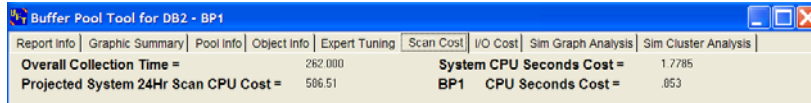
60

**Several sets of information are available in the new version, and some existing information sets have been expanded.**

**Note the IO Delay Secs information, both for the collection period, and a 24 hour projection.**

***Responsive Systems – Performance Software that Works!!***

BPTV8.3 expands existing presentation data with 24 hr projections



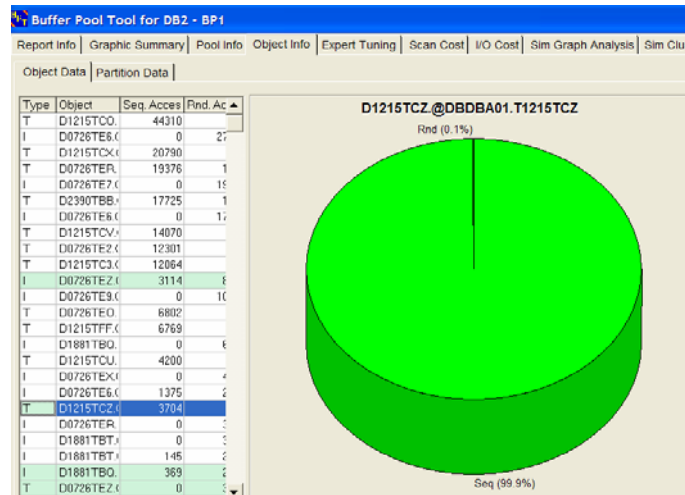
The screenshot shows a window titled "Buffer Pool Tool for DB2 - BP1" with a menu bar containing: Report Info, Graphic Summary, Pool Info, Object Info, Expert Tuning, Scan Cost, I/O Cost, Sim Graph Analysis, and Sim Cluster Analysis. The main display area shows the following data:

Overall Collection Time =	262.000	System CPU Seconds Cost =	1.7785
Projected System 24Hr Scan CPU Cost =	506.51	BP1 CPU Seconds Cost =	.053

**Projecting a 24 hour CPU cost for Sequential Scan in your system.**

## ***Responsive Systems – Performance Software that Works!!***

### **BPTV8.3 Partition Analysis – highlights objects**



© Responsive Systems 2006

[www.respondivesystems.com](http://www.respondivesystems.com)

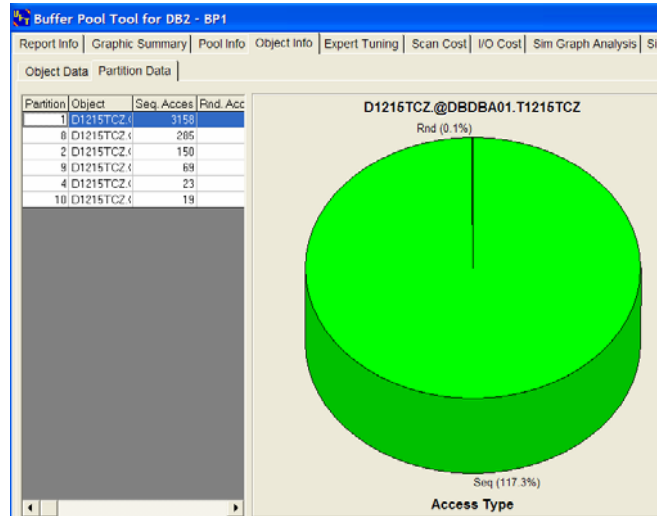
62

**The Object Info tab highlights objects that are partitioned. Selecting an object, then clicking the Partition Data tab takes you the panel that shows you the activity for each individual partition....**

**All columns on this panel can be widened to see the full object name, and all column headings can be “clicked” to sort data into ascending or descending order.**

***Responsive Systems – Performance Software that Works!!***

**BPTV8.3 Partition Analysis – Partition data**



© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

63

**Access and performance data can easily be obtained at the Partition level.**

**Responsive Systems – Performance Software that Works!!**

**Tuning your System with  
the  
Buffer Pool Tool for DB2**

**Joel Goldstein**

Responsive Systems Company

joel@responsivesystems.com

[www.responsivesystems.com](http://www.responsivesystems.com)

732-972-1261

© Responsive Systems 2006

[www.responsivesystems.com](http://www.responsivesystems.com)

64

**Based in Morganville, NJ we can be reached through the web, or by calling  
732-972-1261**