Buffer Pool Tuning
The Techniques that Work

Joel Goldstein

Responsive Systems

(732) 972-1261

Email: joel@responsivesystems.com

Web Site
www.responsivesystems.com

Goals

- Reduce elapsed times
- Save CPU cycles
- Optimize memory usage

Elapsed Time

- Transactions
  - Improves productivity
  - Improves client satisfaction
- Batch jobs
  - Achieve your batch window requirements
  - Maximize throughput
- Performance does not improve by itself
  - It only gets worse…..

CPU Cycles

- **Eliminating I/O reduces CPU costs**
  - **The CPU cost for an I/O does not change when you improve DASD performance**
- **Eliminating I/O**
  - **Usually will not reduce the processor “busy rate”**
  - **It may increase the processor “busy rate”**

Memory

- Larger pools – more memory
  - Do not always improve performance
  - May increase system paging and hurt performance
    - » Might look better statistically, but response time will be worse
- Memory is a *system* resource
  - DB2 is not the *only* sub-system on your processor

Problem indicators

- Pool problem indicators
  - IWTH, DMTH
  - SPTH
  - VWQT, DWQT
  - Write engines unavailable
  - Pagins for read/write - paging/second
  - I/O rate/second
    - » Synch I/O
    - » Prefetch
    - » Sum of all
  - System hit ratio

Buffer Pool Hit Ratio

- The System Hit Ratio Calculation is:

  (Getpages - Sum of all Pages Read) / Getpages

- This calculation can be negative…
  - Dynamic prefetch
  - Thrashing

Buffer Pool Hit Ratio

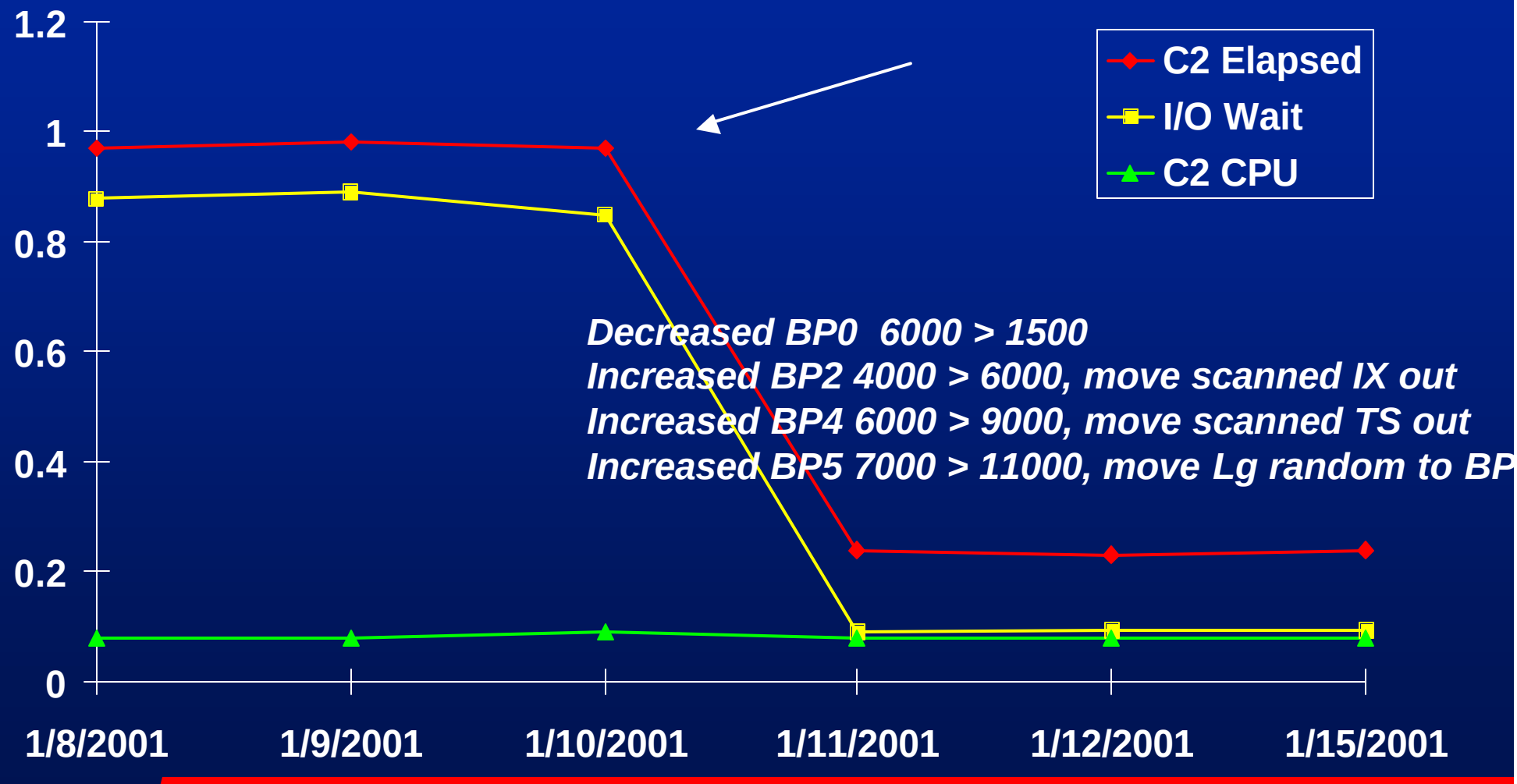
- Application Hit Ratio (GP/RIO) = 97.5%
  - Hey, this sounds like everything is great….
  - Why bother to Tune?
  - This ratio calculation is rarely useful
- System Hit Ratio = 9%
  - Hold on there....
  - This is not good
  - Might be artificially low because of Dynamic Prefetch
  - What is the I/O Rate/Sec?
- HP Effectiveness Ratio

  Pages Read from HP / Pages Written to HP
  - » 1% or 2% is not Good..... 10+% might be....

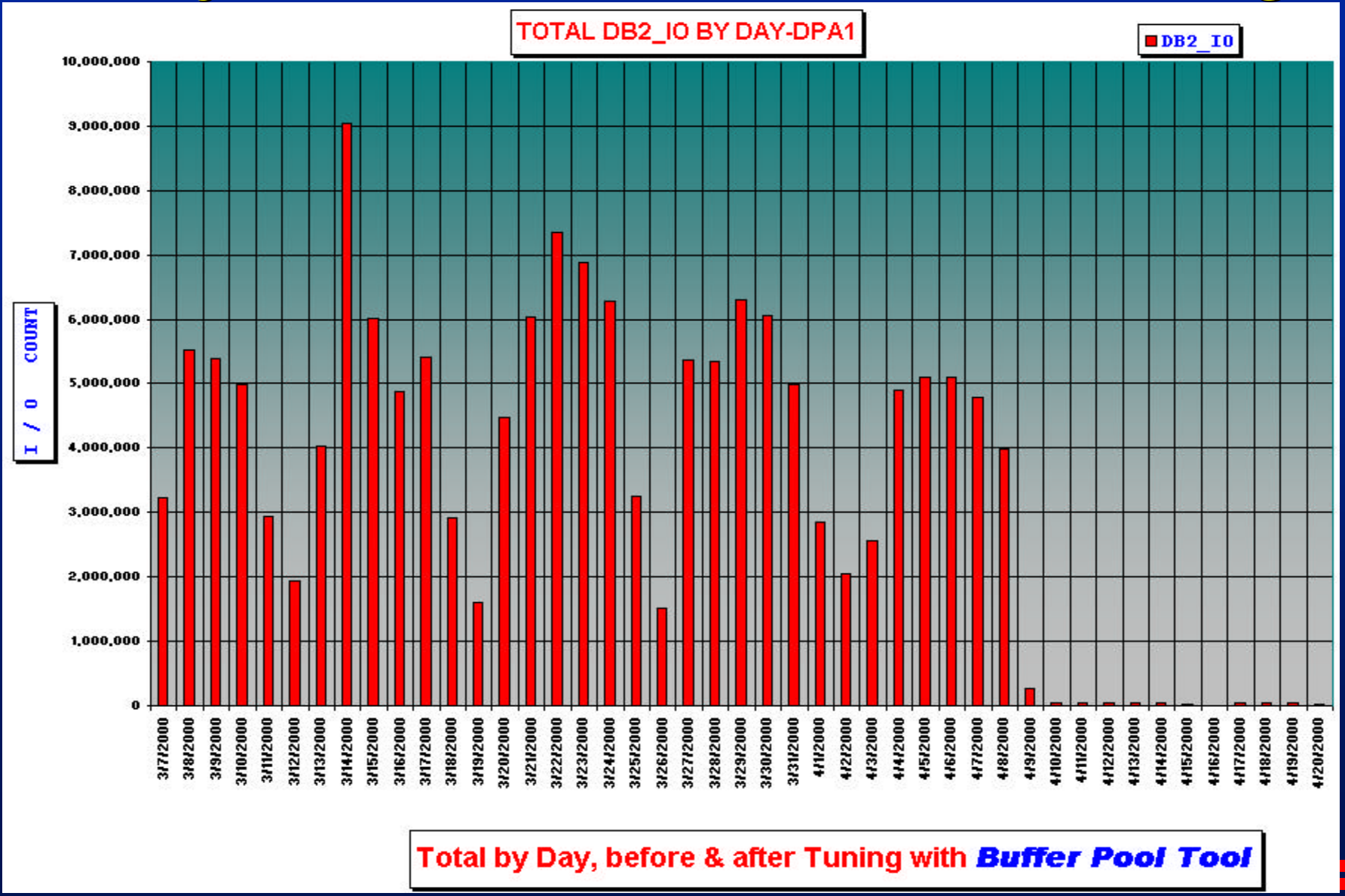
Most Important.....

- I/O rate/second
  - The entire system
  - Each pool
  - Highly active objects
- DASD response times

Application Effects of Buffer Pool Tuning

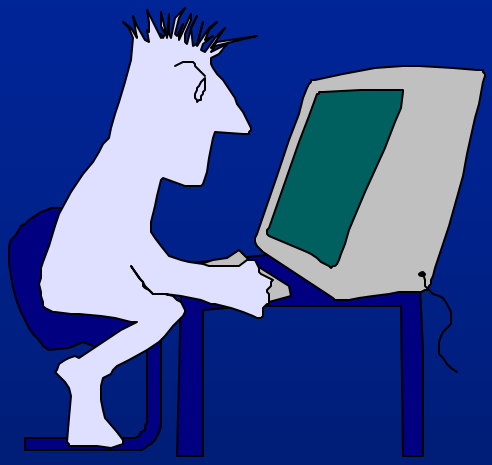


System Effects of Buffer Pool Tuning



Buffer Pool Performance Analysis Beyond the Basic Hit Ratio.....

- **What Data is Necessary?**
  - Detailed Access Information
  - Meaningful Period with your normal workload
  - Meaningful Elapsed Time
    - A few Minutes is NOT Meaningful
  - Must be for the Entire System
  - High volume of trace records
    - SMF or GTF *not viable options*



Analyze the Statistical Data

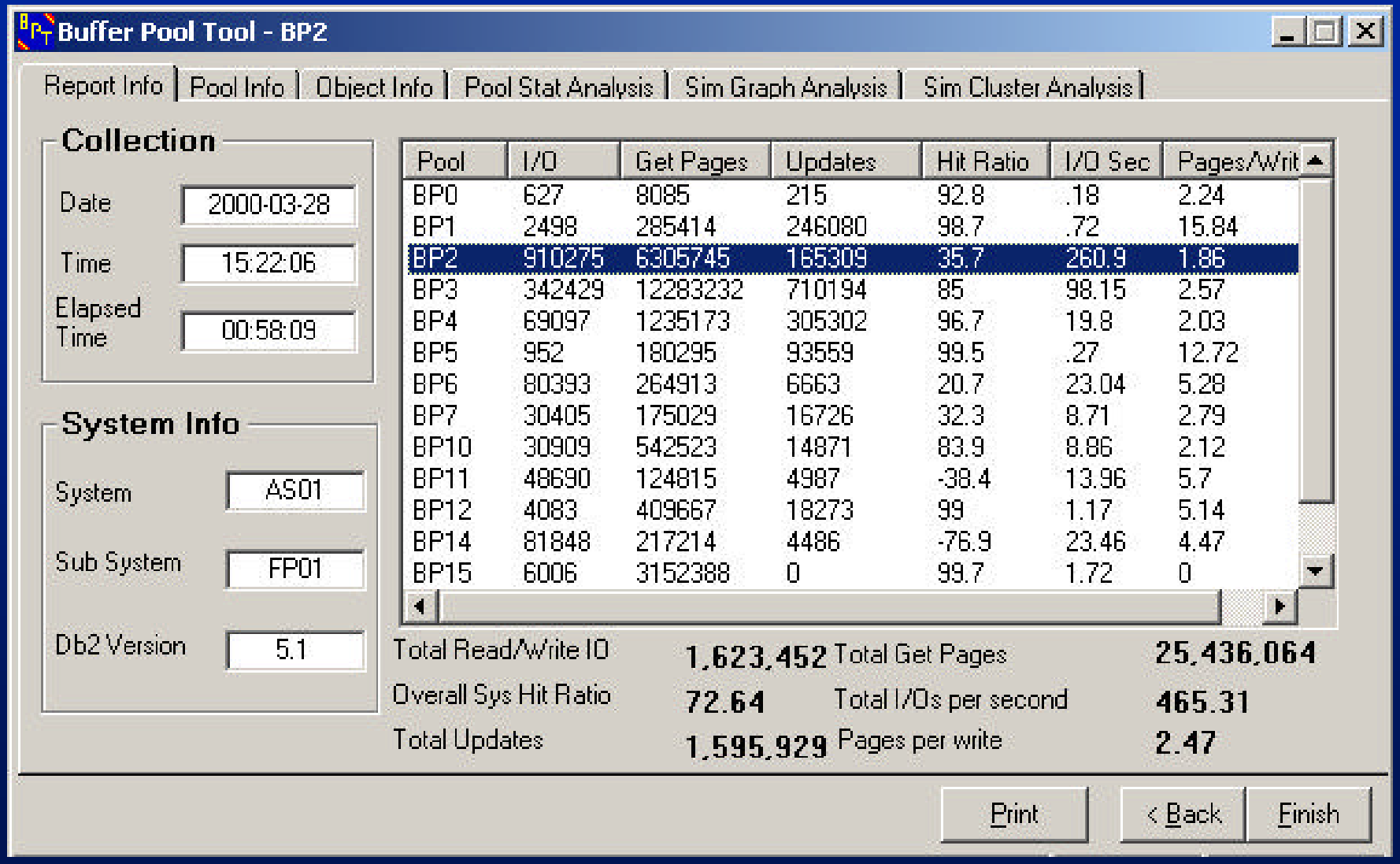
- Indexes not accessed randomly
  - All or mostly Prefetch (SAMOS)
- Objects having highest rates of access, and highest I/O rates
  - Unusual/Unexpected Access/Activity
  - *Watch out for Large Random objects*
- Overall System, Pool, and Object I/O Rates

Predict Performance Gains

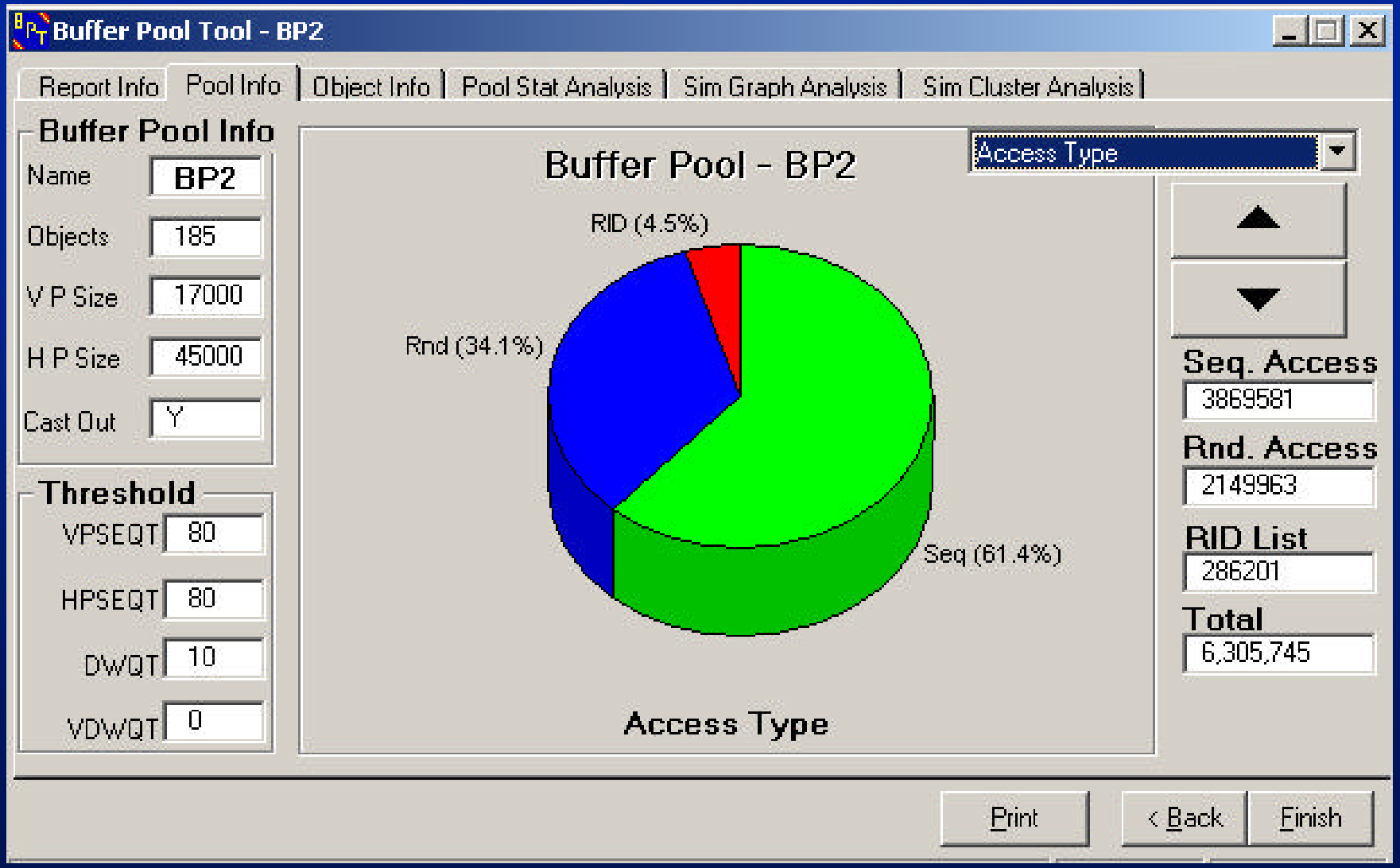
Using Larger Pool Sizes & Moving Objects

- Start at Base Pool Size, and Evaluate Performance Improvement
- Evaluate Average and Maximum Working Sets of Objects in a Pool
- *Grouping similar objects together is the way to optimize both performance and memory requirements (RAMOS, SAMOS)*
  - Proven technique at dozens of companies worldwide
  - Objects can't be moved dynamically from one pool to another
    - » This must be carefully planned and executed to avoid performance problems
  - There will always be trade-offs between OLTP and batch environments

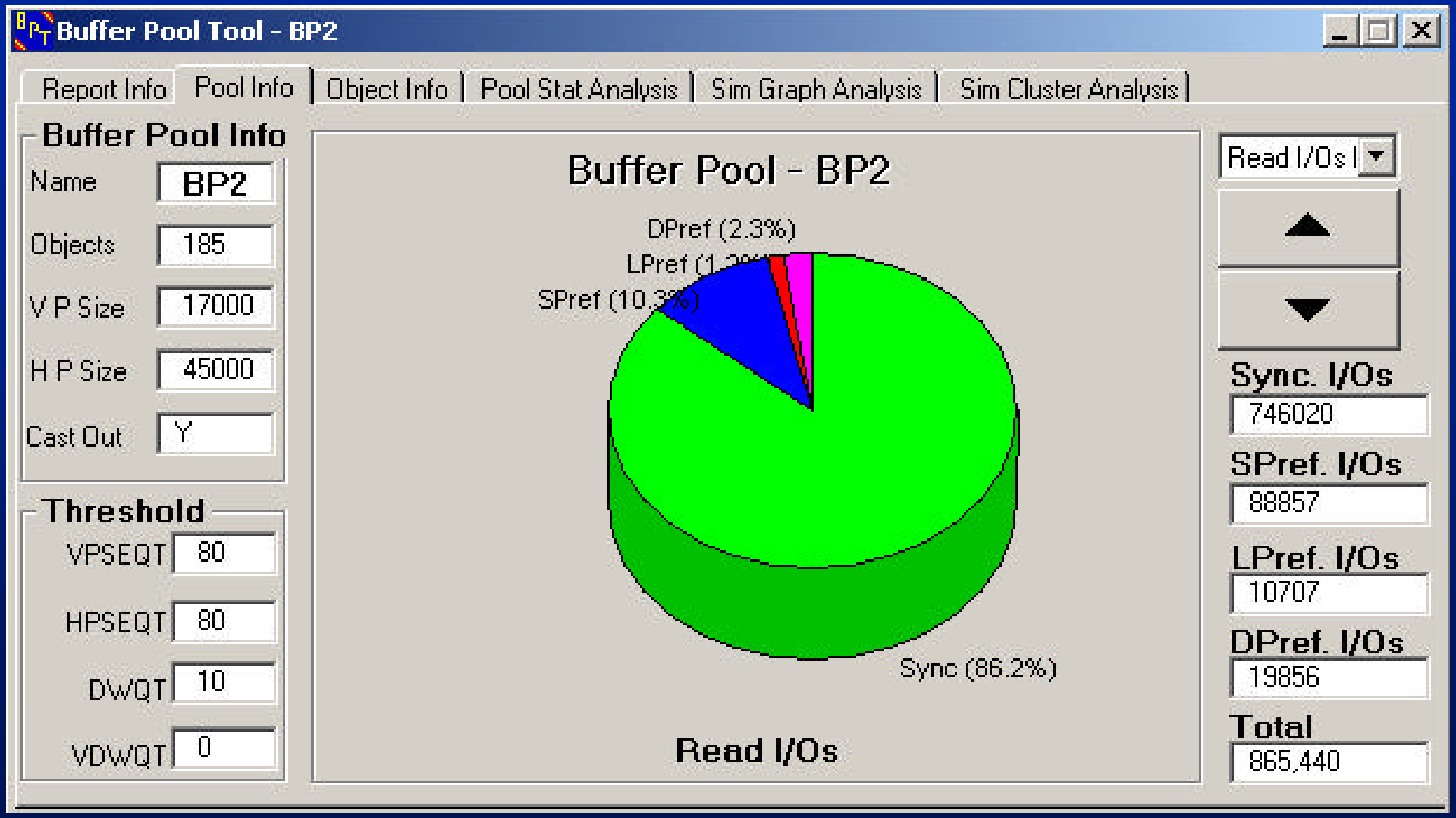
Total System Activity



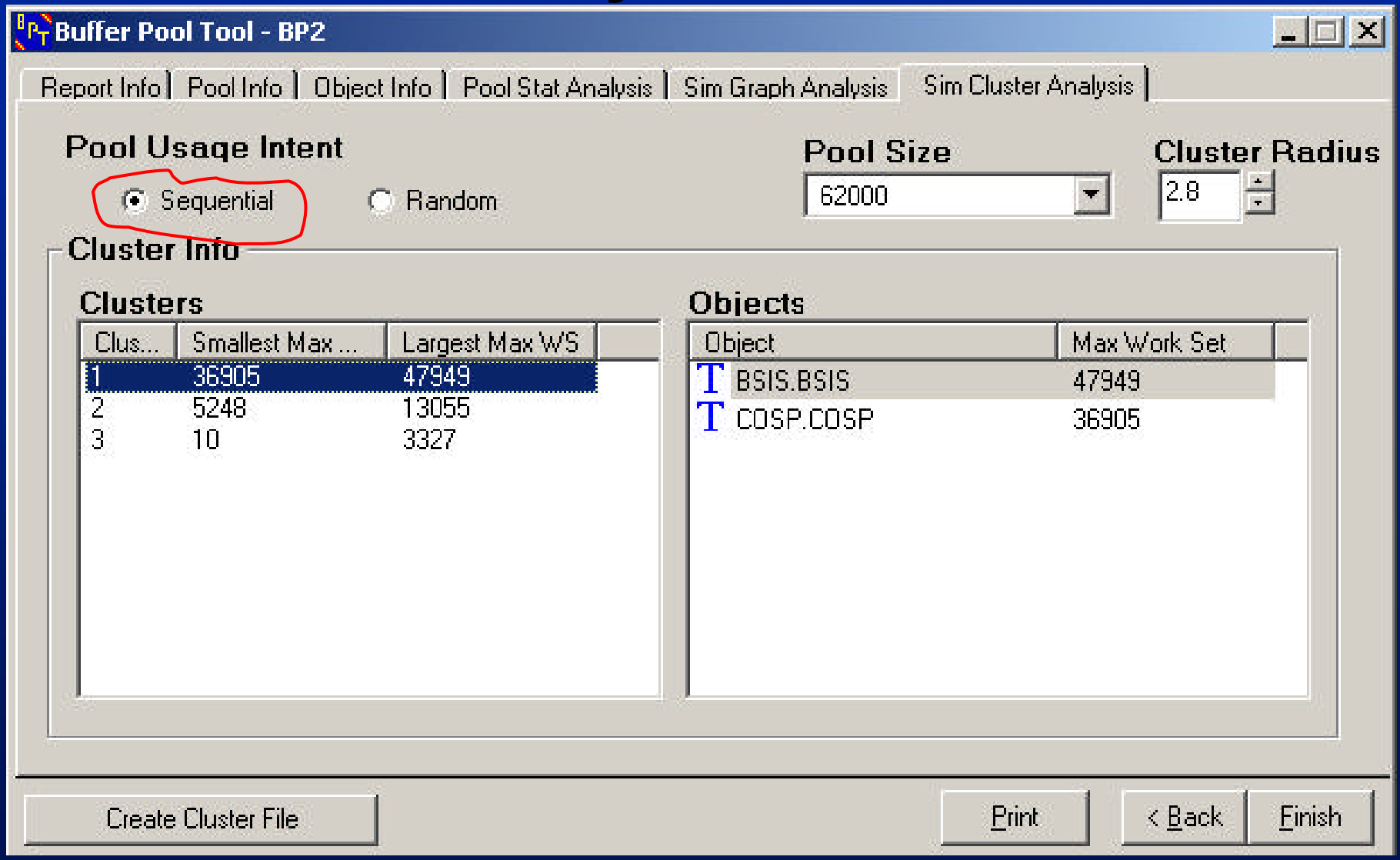
Pool Activity



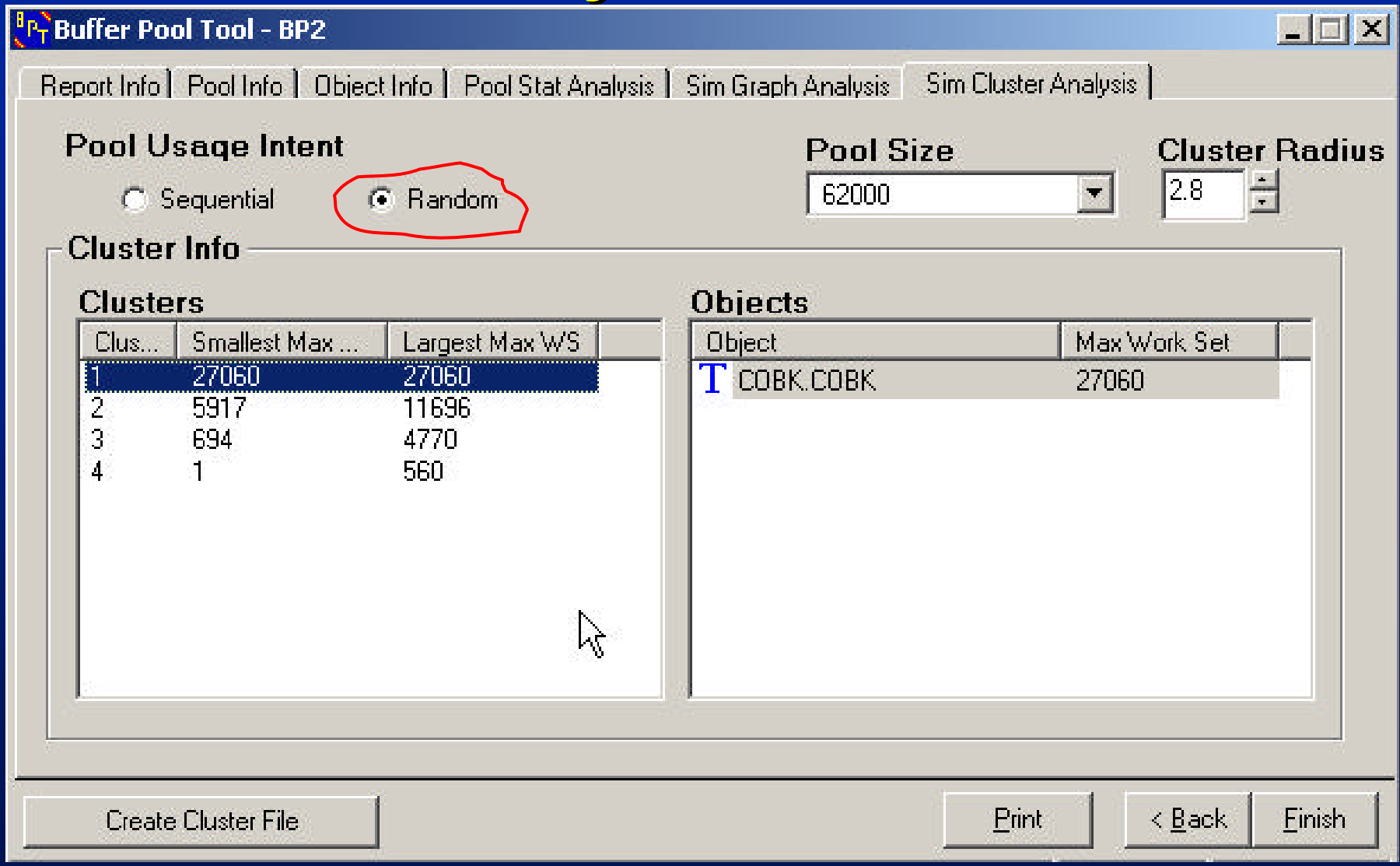
Pool Activity



Pool Activity



Pool Activity



Base Simulation.....Buffer Pool Level

Results of Simulation for Buffer Pool.....BP2
Bpool GetP total.....6,305,745

Bpool Size	GetP used	Num. of Hits
62,000	6,361,468	1,971,692
67,000	6,349,204	2,549,084
72,000	6,322,343	3,007,401
77,000	6,294,862	3,070,251
82,000	6,267,584	3,194,701
87,000	6,233,613	3,244,257

Bpool Size	Pages Read	Read I/O	SyHit Ratio	
62,000	938.5 /S	263.6 /S	35.7 %	<< *Baseline Verificatio*
67,000	912.8 /S	249.0 /S	42.6 %	
72,000	908.2 /S	193.1 /S	46.7 %	
77,000	878.2 /S	175.1 /S	49.9 %	
82,000	826.4 /S	169.2 /S	51.6 %	
87,000	801.3 /S	166.3 /S	52.5 %	

Removing Sequential Objects Prediction.

Results of Simulation for Buffer Pool.....BP2
Bpool GetP total.....3,107,541

Bpool Size	GetP used	Num. of Hits
62,000	2,671,591	2,311,791
67,000	2,583,402	2,267,112
72,000	2,496,415	2,229,453
77,000	2,398,763	2,176,303
82,000	2,277,458	2,132,699
87,000	2,197,131	2,084,518

Bpool Size	Pages Read	Read I/O	SyHit Ratio	
62,000	218.5 /S	101.6 /S	56.2 %	<< - 152.0 I/O Sec
67,000	177.1 /S	84.2 /S	62.6 %	
72,000	145.6 /S	67.3 /S	65.9 %	
77,000	133.2 /S	57.4 /S	69.1 %	
82,000	125.4 /S	48.2 /S	72.5 %	
87,000	121.7 /S	43.2 /S	76.4 %	<< - 223.1 I/O Sec

Compared to base Simulation, removing Large Seq objects saves I/O

Sequential Objects went where?

- Doesn’t matter too much, any sequential pool
  - Separate by small/medium wkset & large wkset
- Sequential objects don’t need a lot of pool space
  - A scan, is a scan…

Buffer Pool Usage Suggestions

- BP0 Catalog & Directory
- BP1 Random Indexes (S/M)
- BP2 Random Indexes (Lg/Huge)
- BP3 Scanned Tablespaces
- BP4 Random Tablespaces (S/M)
- BP5 Random Tablespaces (Lg..)
- BP7 DSNDB07

- BP0 Catalog, Directory
- BP1 Random Indexes (S/M)
- BP2 Scanned Indexes (?)
- BP3 Random Indexes (Lg)
- BP4 Scanned Tablespaces (S/M)
- BP5 Scanned Tablespaces (Lg)
- BP6 Random Tablespaces (S/M)
- BP7 DSNDB07
- BP8 Random Tablespaces (Lg)
- BP9 Random Tablespaces (Transient)

Object Access Characteristics

AMO..S,M,L - Randomly Accessed 'Mostly' SAMO..S,M,L - Sequentially Accessed 'Most

Buffer Pool Usage Suggestions

- Most installations can optimize performance using 6 to 8 Pools
- Some large installations may require a few more...
  - Perhaps up to a dozen
- Too many pools create an Administrative and Performance tracking problem
- Using pools on an application basis, is not a good approach
  - Pools should be a System Resource, not an Application Playground
  - Politics forces many unpleasant situations…

DASD Subsystem - Bottom Up Tuning

- **Every installation has DASD performance problems... it's only a question of magnitude, severity, and time-frame**

- **Causes...**
  - **Not enough physical volumes to spread the I/O workload**
  - **Lack of Staff, Lack of Time**
  - **SMS**
    - **Can do a good job, if you spend the time to *Set It Up Properly* so that *important and critical datasets are on different volumes***
    - **If you just 'give' the DASD environment to SMS, you are almost guaranteed performance problems...**
  - **Mis-Conceptions about new Device/Architectures**
    - **RAID**
      - **Write Penalty, *and* Poor Physical Read**
      - **Can Overload it...**

DASD Subsystem - Bottom Up Tuning

I/O Performance Perspective.....

- < .020 Seconds — 3390
- < .005 Seconds — Cache, Solid State Device
- .000035 Seconds — HiperPool (100+ Times Faster than Cache)

Good Average elapsed times for One 4K Page

How Bad can the Performance be?

```
kdsk    Number of Devices =      961    Number of Samples =      145
        Volser LCU  Rate Util%   Resp = IOSQ + Pend + Conn + Disc  RESV% #Allo
 025E ZTDH15 000    .0    .0     33.6      .0     .3    12.6   20.6     .0  19.0
 027B ZPSH10 000    .0    .0     30.6      .0     .5     1.1   29.0     .0  13.4
 0920 ZTLH06 00E    .0    .2     65.9      .0     .6     4.0   61.2     .0   1.0
 0968 ZTCH21 00F    .0    .1     46.2      .0    1.1    45.1     .0     .0    .0
 096B ZPDHL9 00F    .9   4.3     46.9      .0    1.2     7.0   38.6     .0   3.0
 096D ZTDH86 00F    .2    .9     38.5      .0    1.1     9.8   27.5     .0   4.0
 0970 ZTDH75 00F    .0    .0     51.4      .0    1.0     8.4   42.0     .0  13.0
 0979 ZPDHL8 00F   4.2  19.9     67.6    19.7    1.1     4.3   42.4     .0   3.0
 097A ZTDH81 00F    .0    .1     35.7      .0    3.0     1.6   31.1     .0  12.0
 097B ZTDH82 00F    .0    .1     85.0      .0     .7    21.7   62.5     .0  15.0
```

DASD Subsystem - Bottom Up Tuning

- How do you know when DASD performance is poor?
  - Application Accounting Class 3 data
    - » Average I/O wait
    - » I/O Wait as a percentage of Class 2 Elapsed
  - RMF Reports
    - » ***Beware*** of long interval, and summary reports
      - These tend to mask performance problems
  - DASD/Xpert

DASD Subsystem - DASD/Xpert

Volume	IO Intensity	IO Chit%	IO Resp	IO Conn	IO Pend	IO Disc	IO Queue	IO Count	Cache Cand	Cache Hits	IO/Sec	Cache Ratio
DBP008	2.9974	99	24	8	0	12	4	3034910	3025677	3015931	124.8934	99
DBP002	0.4558	98	44	12	0	11	21	65259	64500	64274	10.3586	99
DBP013	0.2495	94	8	3	0	1	4	112278	108344	105078	31.1883	96
DBP004	0.0577	94	15	2	0	5	8	107255	102179	100384	3.8443	98
MHP052	1.9055	93	21	17	0	1	3	489985	458351	454551	90.7380	99
DBP027	0.1503	93	20	8	0	5	7	20291	19774	18934	7.5152	95
DBP001	0.6072	91	46	11	0	15	20	748404	737134	681341	13.1994	92
DBP016	0.1366	90	23	3	0	6	14	26727	24533	24021	5.9393	97
DBP009	0.6324	89	39	10	0	23	6	175118	163559	156724	16.2146	95
DBP028	0.0253	87	12	7	0	2	3	193384	182658	167636	2.1066	91
DBP021	0.2186	84	18	6	0	6	6	896185	866497	756204	12.1434	87
DBP014	0.1129	82	35	13	0	15	7	43549	38255	35765	3.2259	93
DBP025	1.3510	81	36	13	0	10	13	168875	166694	136379	37.5278	81
MHP029	0.0777	80	36	19	4	4	9	157300	131182	125192	2.1578	95
DBP024	0.3843	80	28	15	0	11	2	123514	104678	98895	13.7238	94
MHP018	0.6779	78	22	19	0	1	2	665549	533097	519383	30.8125	97
DBP019	0.0456	78	13	4	0	5	4	501654	477436	390865	3.5056	81

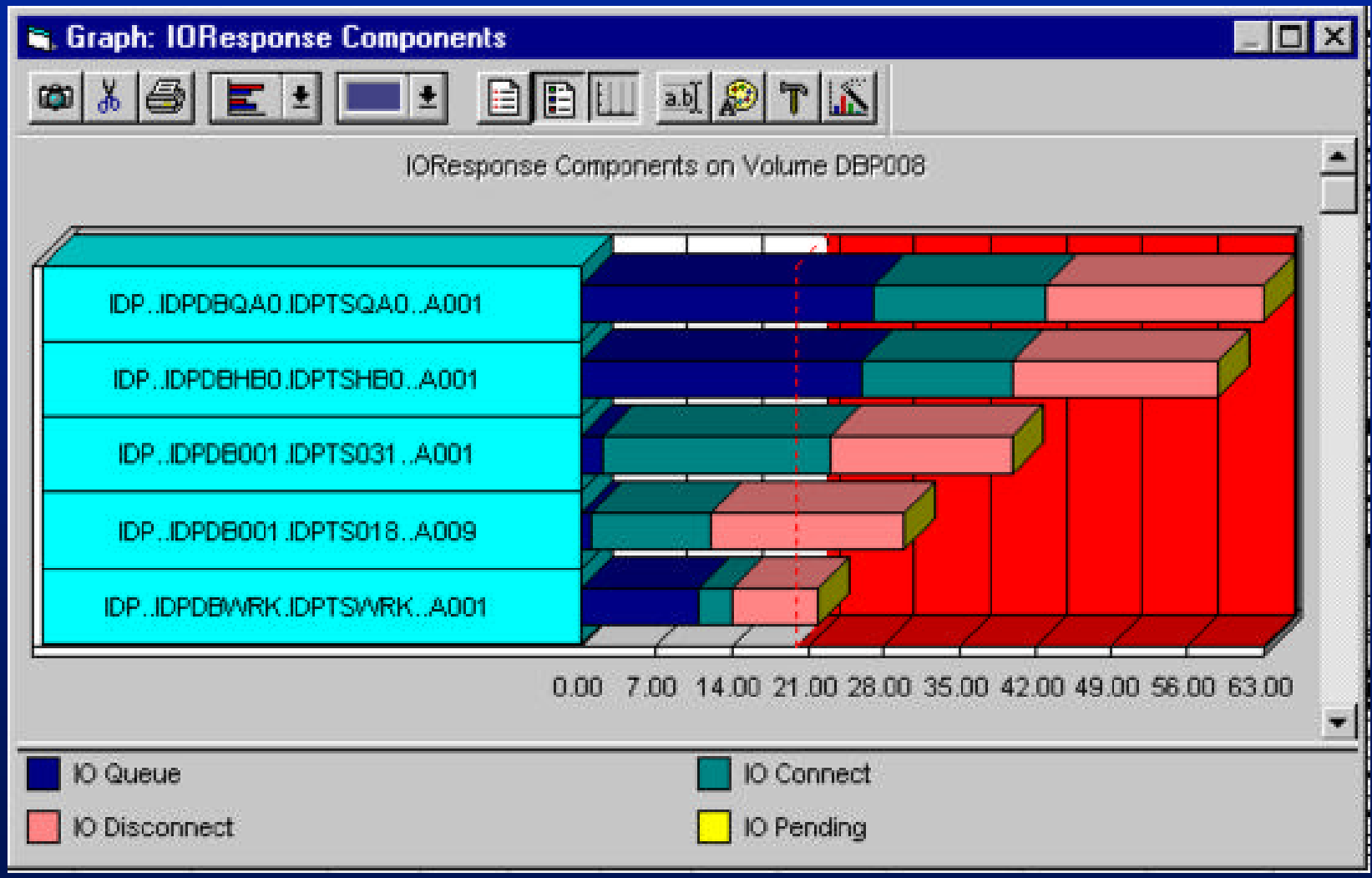
DASD Subsystem - DASD/Xpert

tasets on volume DBP008:

Database	Object	File	Partition	MPL	IO Chit%	IO Resp	IO Conn	IO Pend	IO Disc	IO Queue	IO Count	Cache Cand	Cache Hits
DPDBDL1	IDPTSDL1	I0001	A001	36.6323	100	11	9	0	0	2	2997190	2996348	29923
DPDB002	IDIXRM02	I0001	A001	0.1178	51	17	1	0	13	3	6238	6238	32
DPDBPA0	IDPTSPA0	I0001	A001	0.1074	83	9	3	0	2	4	10742	9868	88
DPDB999	IDPTS999	I0001	A001	0.1072	82	19	4	0	11	4	5078	4623	41
DPDB001	IDPTS031	I0001	A001	0.0314	22	40	21	0	17	2	707	206	1
DPDBWRK	IDPTSWRK	I0001	A001	0.0214	67	22	3	0	8	11	876	786	5
DPDB001	IDPTS018	I0001	A009	0.0227	6	30	11	0	18	1	6130	458	3
DPDBFIN	IDPTSFIN	I0001	A001	0.0411	78	14	8	0	3	3	2640	2198	20
DPDB003	IDPTS045	I0001	A001	0.0325	74	10	2	0	4	4	2926	2828	21
DPDBBIO	IDPTSBIO	I0001	A001	0.0170	87	9	4	0	2	3	1704	1558	14
DPDBHA0	IDPTSHA0	I0001	A001	0.0038	85	10	4	0	2	4	338	303	2
DPDBNRS	IDPTSNRS	I0001	A001	0.0057	67	18	8	0	6	4	283	227	1
DPDBHB0	IDPTSHB0	I0001	A001	0.0016	16	59	14	0	19	26	25	15	
DPDBQA0	IDPTSQA0	I0001	A001	0.0013	6	63	16	0	20	27	18	9	

Moving the high volume object will only move the problem somewhere else

DASD Subsystem - DASD/Xpert



DASD Subsystem - DASD/Xpert

Volume	IO Intensity	IO Chit%	IO Resp	IO Conn	IO Pend	IO Disc	IO Queue	IO Count	Cache Cand	Cache Hits	IO/Sec	Cache Ratio	Write Cand	Write Hits	Write Ratio
DSP003	250.2102	61	1283	8	0	5	1270	526553	324101	319737	195.1963	98	127166	123814	97
DBP008	2.9974	99	24	8	0	12	4	3034910	3025677	3015931	124.8934	99	9301	9139	98
MHP052	1.9055	93	21	17	0	1	3	489985	458351	454551	90.7380	99	780	779	99
DBP025	1.3510	81	36	13	0	10	13	168875	166694	136379	37.5278	81	7082	7054	99
MHP018	0.6779	78	22	19	0	1	2	665549	533097	519383	30.8125	97	3339	3331	99
DBP007	0.6459	61	20	5	0	11	4	1249775	814894	762099	32.2939	93	1159	1156	99

Shoot yourself in the foot...

Datasets on volume DSP003:

Database	Object	File	Partition	IO Intensity	IO Chit%	IO Resp	IO Conn	IO Pend	IO Disc	IO Queue	IO Count	Cache Cand	Cache Hits	IO/Sec	Cache Ratio	Write Cand	Write Hits
DSNDB07	DSN4K03	I0001	A001	418.9352	54	1598	9	0	6	1583	235946	129520	127181	262.1622	98	51010	49143
DSNDB07	DSN4K01	I0001	A001	175.8038	66	1125	8	0	5	1112	140643	94239	92918	156.2700	98	41653	40534
DSNDB07	DSN4K02	I0001	A001	187.4550	66	1125	8	0	4	1113	149964	100342	99638	166.6267	99	34503	34137

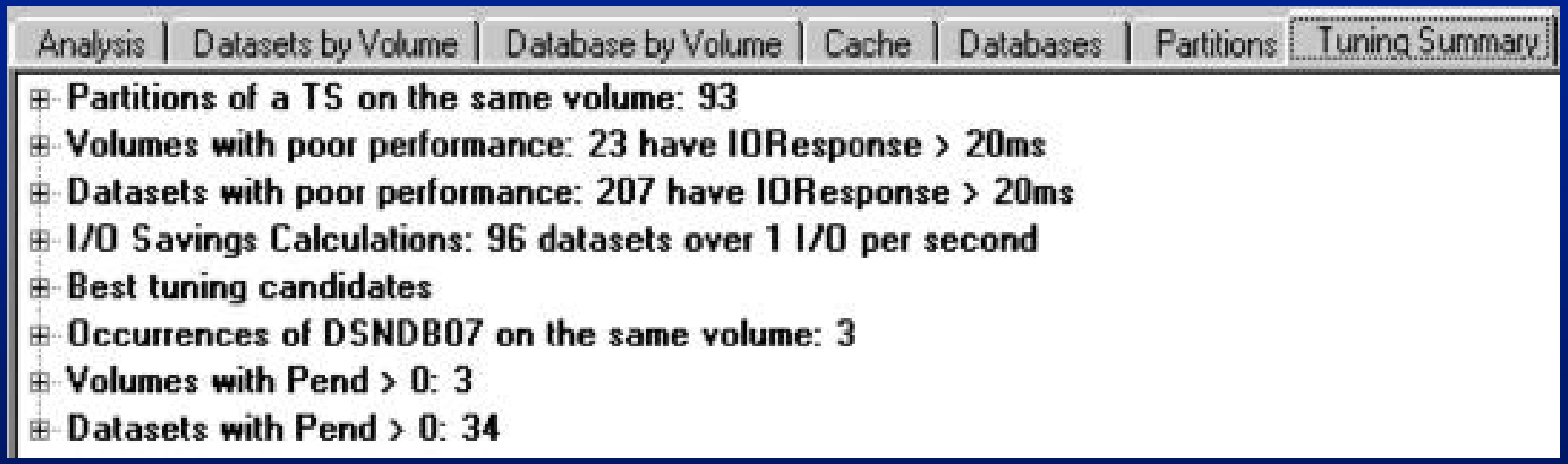
DASD Subsystem - DASD/Xpert

Partition Analysis

IDP.DSNDBD.IDPDB001.IDPTS018.I0001

Partition	Volume	IO Resp	IO Conn	IO Pend	IO Disc	IO Queue	IO Count	Cache Cand	Cache Hits	IO/Sec
A001	DBP020	49	31	0	16	2	320	32	29	0.1185
A002	DBP020	49	31	0	16	2	685	74	71	0.2537
A003	DBP020	43	24	0	17	2	1855	148	144	0.5153
A004	DBP024	55	30	0	23	2	6262	124	120	1.7394
A005	DBP018	55	21	0	29	5	6962	629	624	1.5471
A006	DBP020	37	16	1	18	2	6835	545	536	1.2657
A007	DBP009	47	16	0	30	1	7474	699	688	1.3841
A008	DBP011	43	20	0	22	1	8094	164	133	1.4989
A009	DBP008	30	11	0	18	1	6130	458	388	0.7568
A010	DBP005	32	11	0	19	2	6601	509	368	0.8149
A011	DBP025	55	20	0	15	20	30	12	4	0.0111

DASD Subsystem - DASD/Xpert



DASD Subsystem - DASD/Xpert

DASD Subsystem - Bottom Up Tuning

- Identify the objects with:
  - High I/O Count
  - High I/O Intensity
  - High I/O Cache Hit Rate
    Cache Hits/ I/Os * 100
- These are the Objects that provide Opportunity for BP Tuning
  - Eliminate the I/O
  - Eliminate the Application I/O Wait Time
  - Save the CPU Cost of the I/Os
- DASD/Xpert
  - Highlights the greatest paybacks
  - Recommends which objects to move to a different volume

Our Solutions: Buffer Pool Tool

- Established industry standard
- The only product available today that can Predict…
  - The proven track record, with client success stories
- Works for DB2 version 3 thru version 7
- Simple install, up and running in 20 minutes
- Extensive utilities

Our Solutions: Buffer Pool Tool

- PC components
  - Easy to install, WIN98, NT, WIN2000
  - Extensive graphics
  - Tuning suggestions
  - Coupling Facilty sizing component
- Version 7 available NOW!!!

Our Solutions: Buffer Pool Tool

- Compared to the competition….

Our Solutions: Buffer Pool Tool

- **Bigger**
  - More extensive
  - More utilities, more options

Our Solutions: Buffer Pool Tool

- **Better and Faster**
  - **Runs quicker**
  - **Less overhead**

Our Solutions: Buffer Pool Tool

- Direct email and telephone access to support
  - Talk directly with the people who really understand the software
  - Speak with the people who really know how to tune *your* DB2 system

Our Solutions: Buffer Pool Tool

- **Much less expensive**
  - **It is your job to provide the best solution for your company, at the lowest cost….**

Our Solutions: Buffer Pool Optimizer

- Runs all possible simulation/prediction scenarios, provides:
  - Proper sizing for all pools within memory constraints
  - Proper grouping of objects across the pools
  - The alter statements to implement all the changes
- Automates the tuning process
- You decide when to implement….

(requires Buffer Pool Tool as a base)

Our Solutions: IMS Buffer Pool Tool

- Extends our *industry standard* tuning technology to the IMS database marketplace

Our Solutions: Clamp

- Combines additional DB2 Catalog information with Buffer Poo Tool statistics
  - Provides in-depth analysis of object access and index usage
  - Helps you determine which indexes are not used, and not needed
  - Shows you how the indexes are really used

(requires Buffer Pool Tool as a base)

Our Solutions: DASD/Xpert

- Designed for the DB2 analyst
- Makes it easy to find and analyze DASD performance problems
- Inexpensive and easy to use

Summary.....

- There is a huge amount of performance data available in the Statistics and Accounting Records
  - Learn how to use it - Find your problems. *There are more than you may think..*
  - Be *PROACTIVE* about tuning - Don't wait for the phone to ring
- Buffer Pool Tuning is a Science, not a Mystical Art
  - Software does exist - today, that can do this.... the ***Buffer Pool Tool***
  - **Optimizer** automates the process and reduces your staff workload
- DASD Tuning is <u>still</u> <u>a</u> real <u>Challenge</u> - with Big Paybacks
  - **DASD/Xpert** shows you the problems, and provides recommendations

Let the Tuning Begin .....

Questions ?

Demos of the PC components of:

Buffer Pool Tool for DB2

DASD/Xpert

Clamp

Our Booth in the Vendor area

Also, find out about *Buffer Pool Optimizer, IMS Buffer Pool Tool*