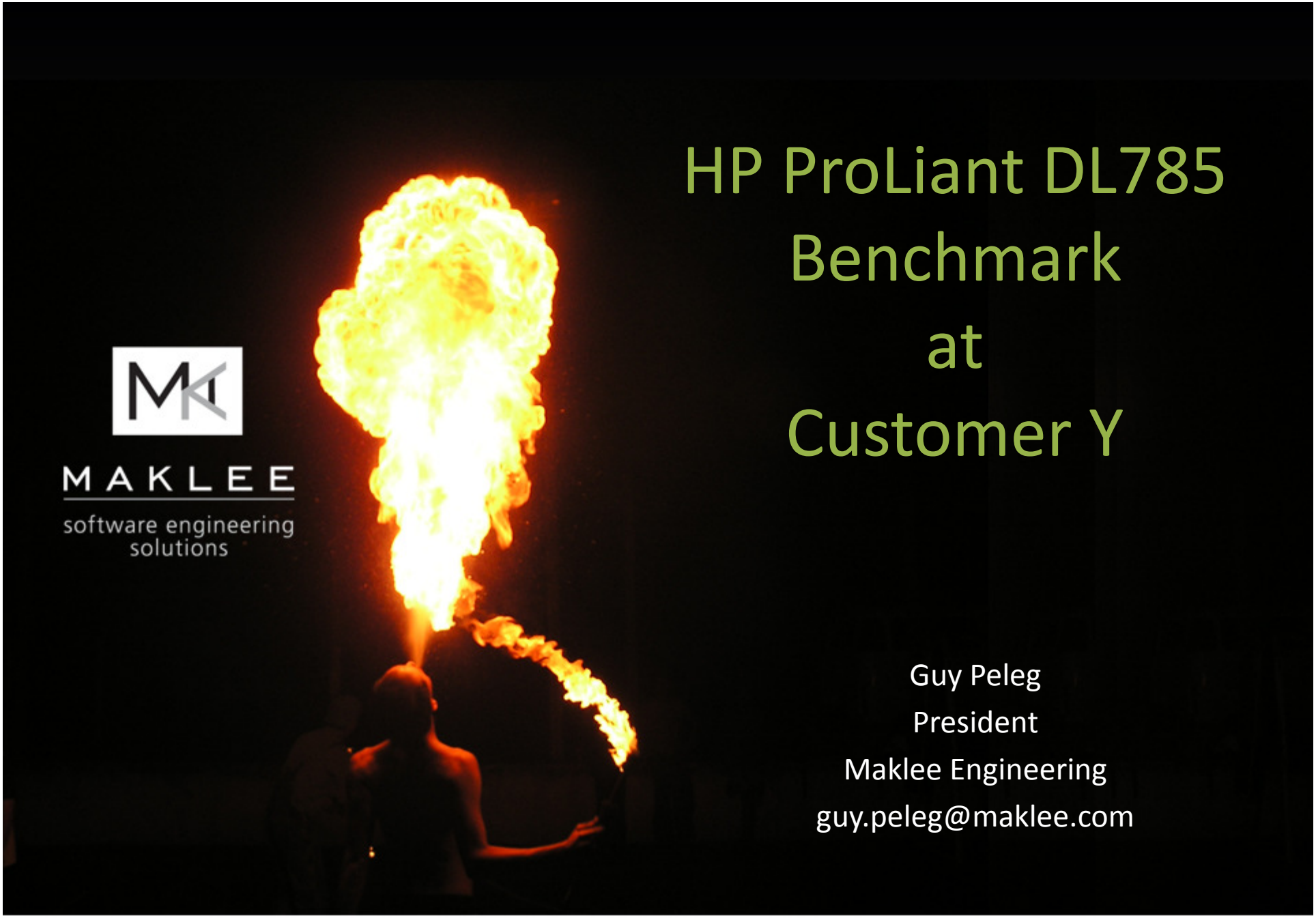
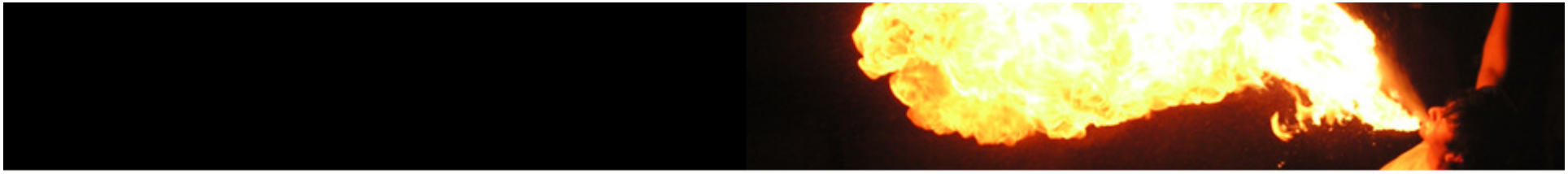


HP ProLiant DL785 Benchmark at Customer Y

MAKLEE

software engineering solutions

Guy Peleg
President
Maklee Engineering
guy.peleg@maklee.com



*This is a **censored** version of a presentation highlighting results of an Oracle benchmark on HP DL785 running Red Hat Linux*

MAKLEE

Benchmark Goals

- Benchmark the XYZ application stress test on HP ProLiant DL785.
- Tune the environment to achieve optimal results.
  - Performance is measured by the number of transactions per second.
- Prove that HP ProLiant DL785 is capable of handling production load and can provide headroom for future growth.

MAKLEE

Configuration

- Hardware
  - HP ProLiant DL785 G5 Server.
  - 8 Quad core AMD Opteron 8393 SE processors
    - Total of 32 cores
    - 3.1 GHz
  - 512GB Physical memory
  - Tier 1 customer Y storage
    - 250GB Volume
- Operating System
  - Red Hat Enterprise Linux 5 update 4
    - 2.6.18-164.el5
    - 64-bit Kernel
- Database
  - Oracle 10.2.0.4

MAKLEE

Benchmark Rules

- Commit 10 rows per transaction.
  - Emulate OLTP load.
- The benchmark is driven by 40 clients.
  - Client processes distributed between 2 Sun development servers.
  - Client processes evenly distributed between Sun servers (20 processes per server).
- The database SGA is limited to 32GB.
- Streams replication turned off.
- Avoid changing SQL statements.
  - Restrict all changes to O/S and Database.

MAKLEE

”Out of the box” Results

- Using ”Out of the box” configuration the HP ProLiant DL785 server performed 44,000 transactions per second.
  - ***2.64 Million Transactions Per Minute.***
- CPU Utilization was 18%
  - 5.7 CPUs utilized.

MAKLEE

Database Tuning

- Maklee was enaged to assist tuning the Oracle database.
- Maklee's approach for Database tuning include changes in the following areas:
  - Operating System parameters
  - Database parameters
  - SQL statements

MAKLEE

Database Tuning

- Maklee's tunning recommendations increased the benchmark's throughput by 20% to 53,311 transactions per second
  - ***3.19 Million Transactions per Minute***
- CPU Utilization was 20%

At this point:

- *The system was not able to perform additional work. Waiting for transactions to be written to the redo log file.*
- *We hit the limit on the number of synchronous I/O operations per second.*
  - *1ms – 3ms for write latency is too slow for our workload, resulting in ~ 6ms wait time per commit.*
- *The system was performing a total of 30,000 disk operations per second.*

MAKLEE

I/O Speed-up

- We needed a way to speed up redo log writes.
- Customer Y should move redo log files of databases with high-end workloads to Fusion-io drives.
- Such drives were not available during our benchmark.
- Therefore we chose emulating Fusion-io by turning the database commits to Asynchronous operations.
- Writes to the redo log file still occurs, but it happens in the background.
- Lowers elapsed time of writing redo data from 6 milliseconds to under 1 millisecond.
- The I/O speed up made Maklee's tuning more efective.

– ***Maklee later verified these results with a real Fusion-io drive***

MAKLEE

Database Tuning – Final Results

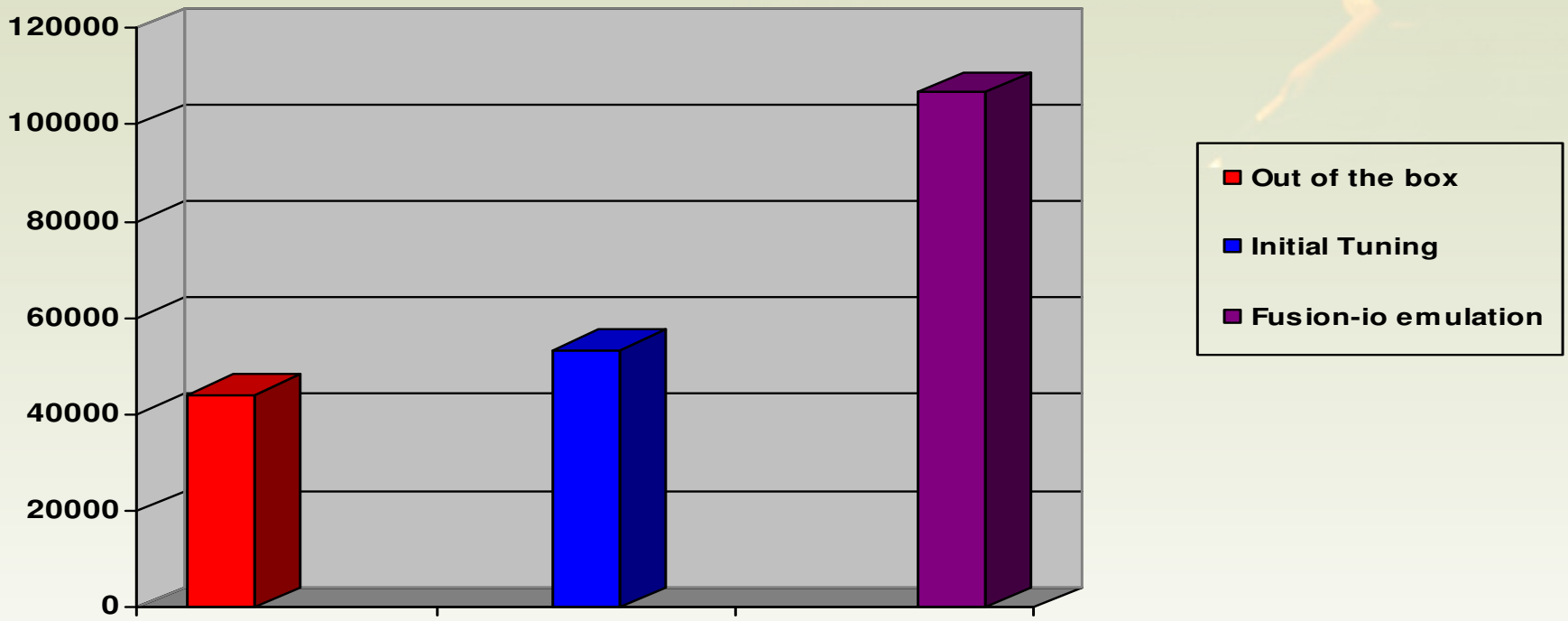
The system now performed 106,705 transactions per second

- ***6.4 Million Transactions per Minute***

- CPU Utilization was 45%
- The system was performing a total of ~ 70,000 disk operations per second.
- The next slide summarizes the results achieved during the benchmark.

MAKLEE

XYZ Benchmark Results



Transactions per second
More is better

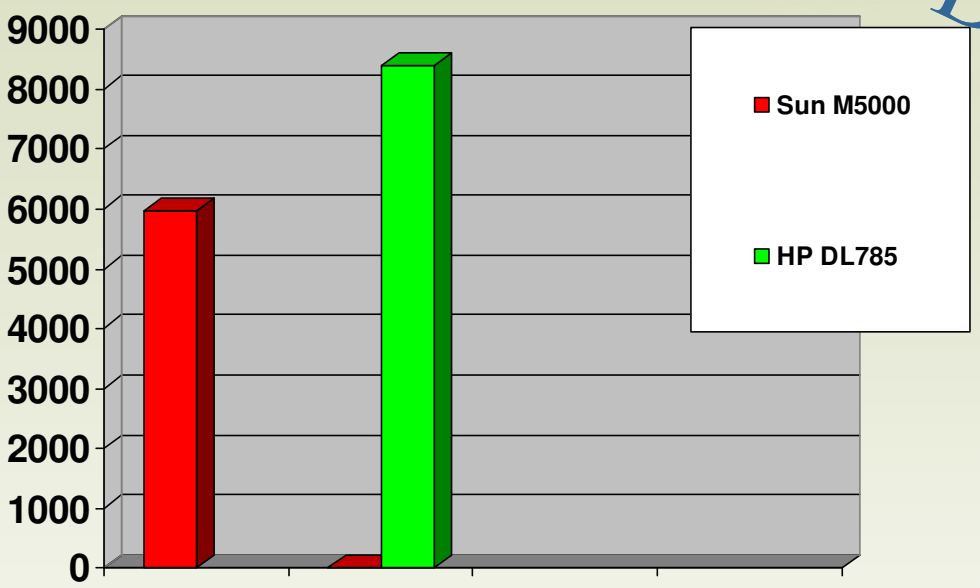
MAKLEE

M5000 vs. DL785

- Customer Y currently runs the XYZ application on Sun SPARC M5000 servers.
- We compared the performance of key SQL statements used by XYZ while running on the Sun M5000 server and HP DL785 server.
- Note – The HP DL785 results captured during synchronous execution, after tuning the environment.
  - During this run we performed 53K transactions per second.

MAKLEE

Insert into A



HP DL785 is 40% faster

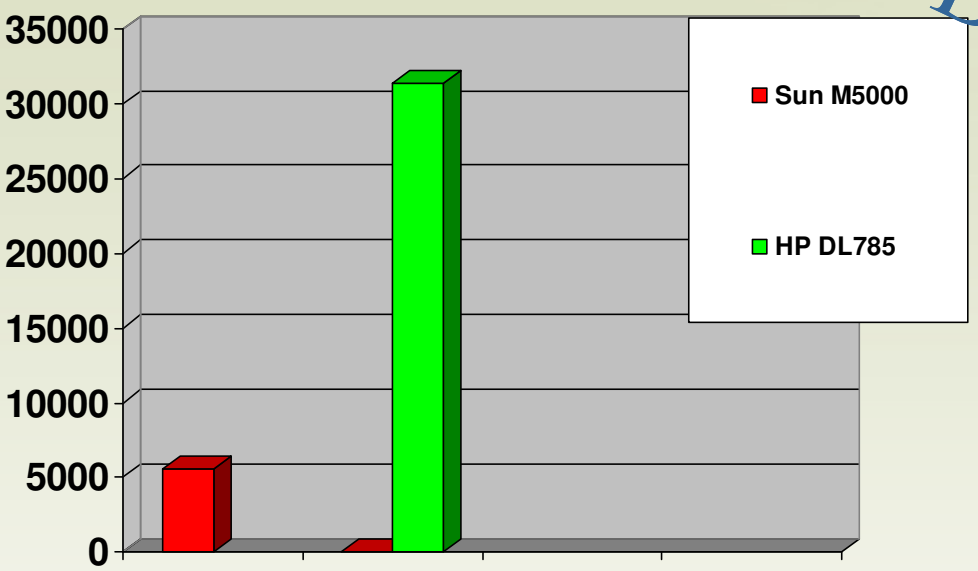
Insert operations per second

(more is better)

```
INSERT INTO A VALUES(:B10 , :B9 , :B8 , :B7 , :B6 , :B5 , :B4 , :B3 , :B2 , :B1 )
```

MAKLEE

Update B



HP DL785 is 5.6X faster

Update operations per second

(more is better)

```
UPDATE B SET C=C-:B3 WHERE D=:B2 AND E=:B1
```

MAKLEE

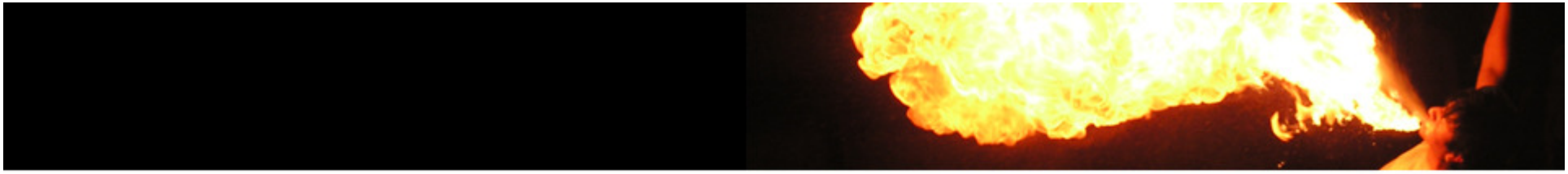
Summary

- The HP ProLiant DL785 achieved outstanding result of 106,705 transactions per second.

- Maklee did not change any SQL statement as part of the tuning project.
  - The benchmark utilizes a very small number of SQL statements (only 3 statements).
  - All statements are very simple with a cost of 1.

- Parameters used for the XYZ benchmark designed to fit the unique nature of the XYZ workload. These parameters should not be applied to other workloads.

MAKLEE



Technical Slides

MAKLEE

O/S changes

- The following Linux parameters were changed:
  - vm.swappiness=0
  - vm.dirty_background_ratio=3
  - vm.dirty_ratio=15
  - vm.dirty_expire_centisecs=500
  - vm.dirty_writeback_centisecs=100
- # echo ‘id –g oracle’ > /proc/sys/vm/hugetlb_shm_group
- Default memory pages size is 4KB.
- Enabled Huge memory (page size is 2MB) .

MAKLEE

Database Changes (1 OF 2)

- Disabled automatic memory management of the SGA
  - SGA_TARGET=0
  - SHARED_POOL_SIZE=1G
  - DB_CACHE_SIZE=2G
  - DB_32K_CACHE_SIZE=28G
  - JAVA_POOL_SIZE=20m
  - LOG_BUFFER=283062272
- More Database parameter changes:
  - DB_WRITER_PROCESSES=4
  - LOG_ARCHIVE_MAX_PROCESS=16
  - _BLOCK_HASH_BUCKETS=16777216
  - _SPIN_COUNT=10000

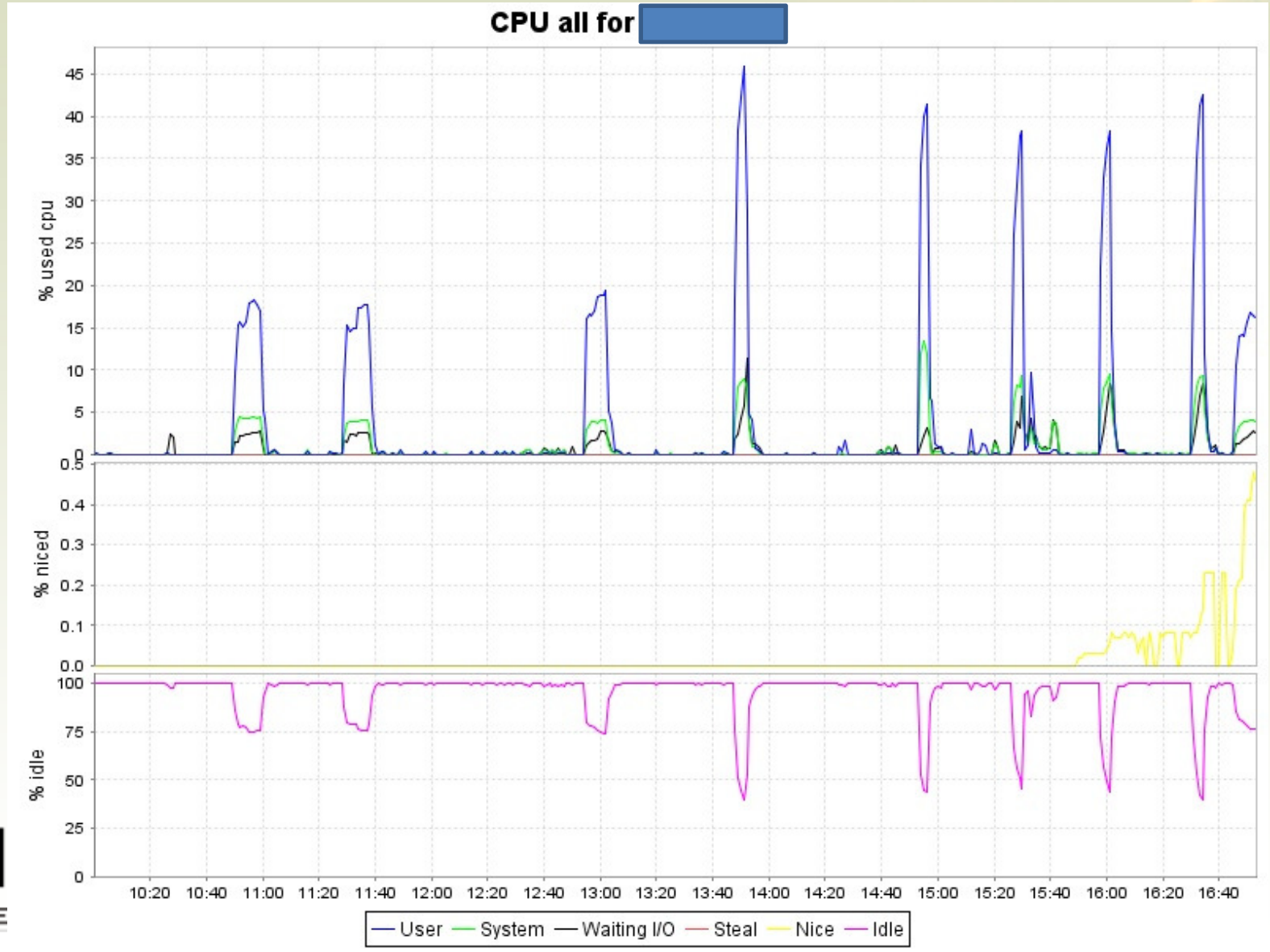
MAKLEE

Database Changes (2 OF 2)

- Redo Log files
  - Eliminated second redo log member in each group.
    - Use hardware mirroring.
  - Increased the size of each log file from 1GB to 5GB.
- Rebuilt all 6 indexes owned by the XYZ schema.
- Asynchronous commits
  - COMMIT_WRITE='BATCH,NOWAIT'
- 32KB tablepspaces
  - Migrated the XYZ data files + indexes into a 32KB tablespace.

MAKLEE

SAR data – CPU Utilization



MAKLE

SAR data – I/O statistics