collective

Oracle for System Administrators

Mark E. Dawson Jr.

Collective Technologies

January 23rd, 2001

collective

Introduction

- Oracle is commonly run on Unix platforms.
- System Administrators are tasked with managing such Unix environments
- Understanding of the interaction of the Unix environment and the database application essential
  - Results in a much better ability in meeting client SLAs (Service Level Agreements).

collective

Oracle

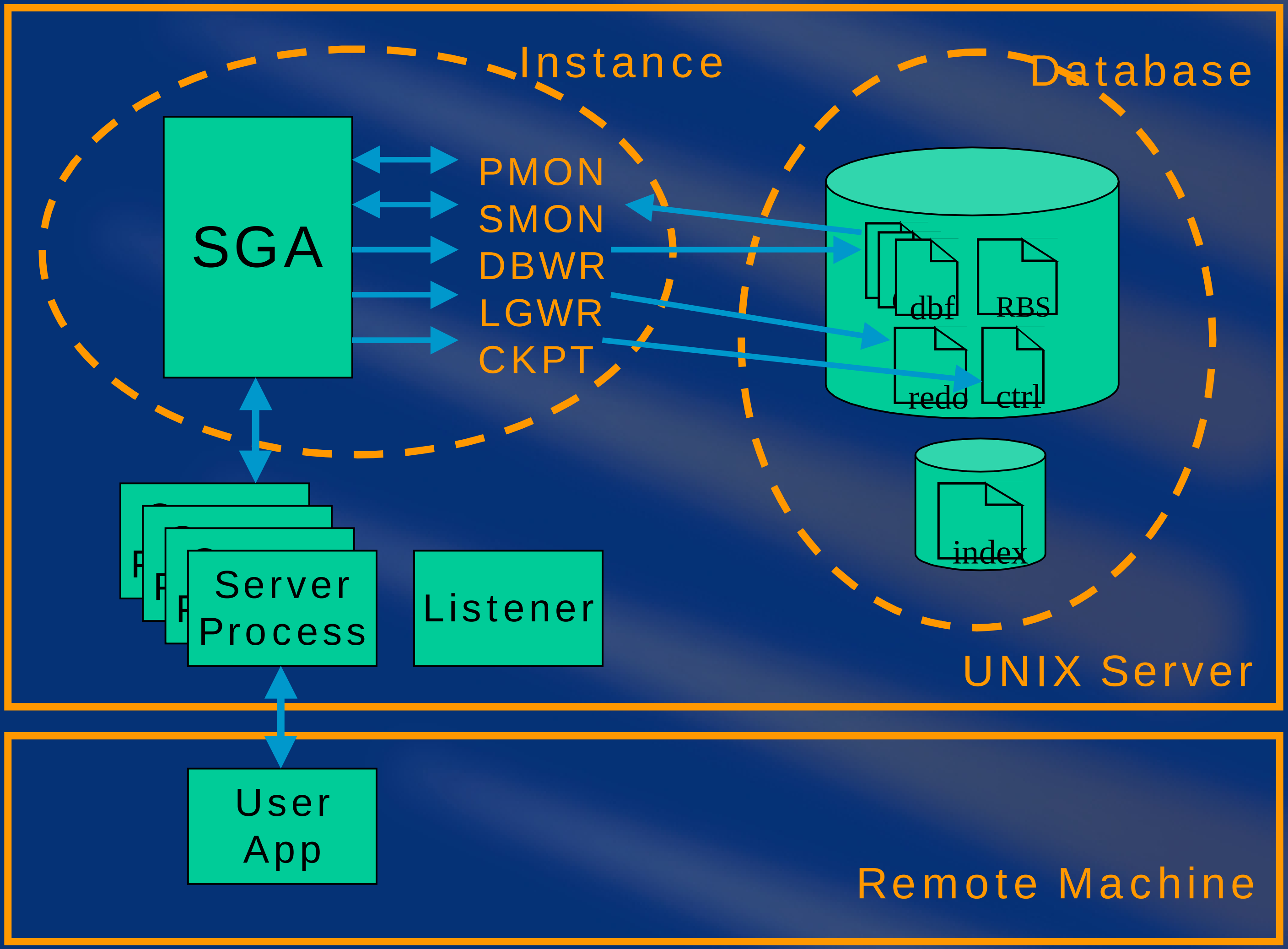
- Purpose
  - Why we use Oracle
- How it Works
  - Instance
  - Database

collective

Purpose of Oracle

- Effectively and reliably manage large amounts of data in a multi-user environment
  - Must accomplish the above while maintaining a high level of performance.
  - Provide efficient solutions for failure recovery and read consistency
  - Provide a high level of data access concurrency

Oracle Overview

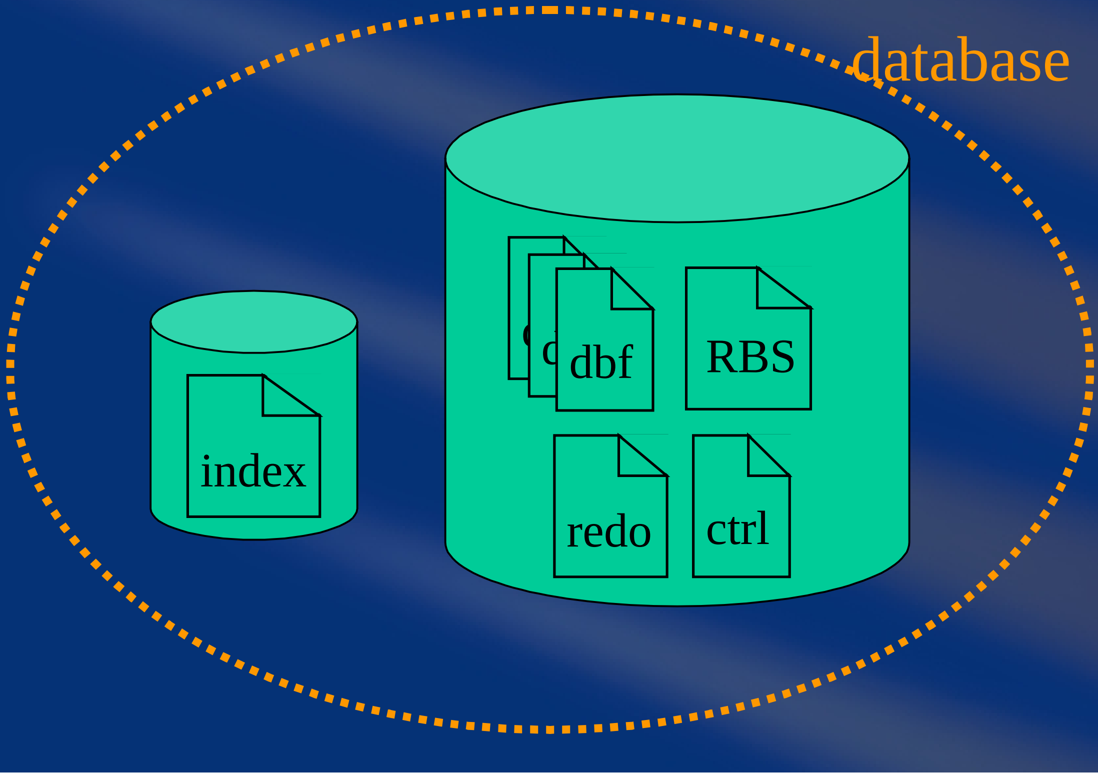


Oracle Components

- Oracle consists of a ***database*** and an ***instance***.
  - A ***database*** includes all the physical data files, control files, and redo log files that will hold your data and Oracle's metadata information.
  - An ***instance*** is a combination of the pool of physical memory (RAM) allocated to Oracle, referred to as the **System Global Area (SGA)**, and the background processes that Oracle spawns to use this memory pool.
    - **SGA:** Area where Oracle attempts to cache database data for faster access (RAM I/O is about 1000x faster than disk I/O).

Oracle Database

- Oracle consists of a ***database*** and an ***instance***.
  - A ***database*** includes all the physical data files, control files, and redo log files that will hold your data and Oracle's metadata information.



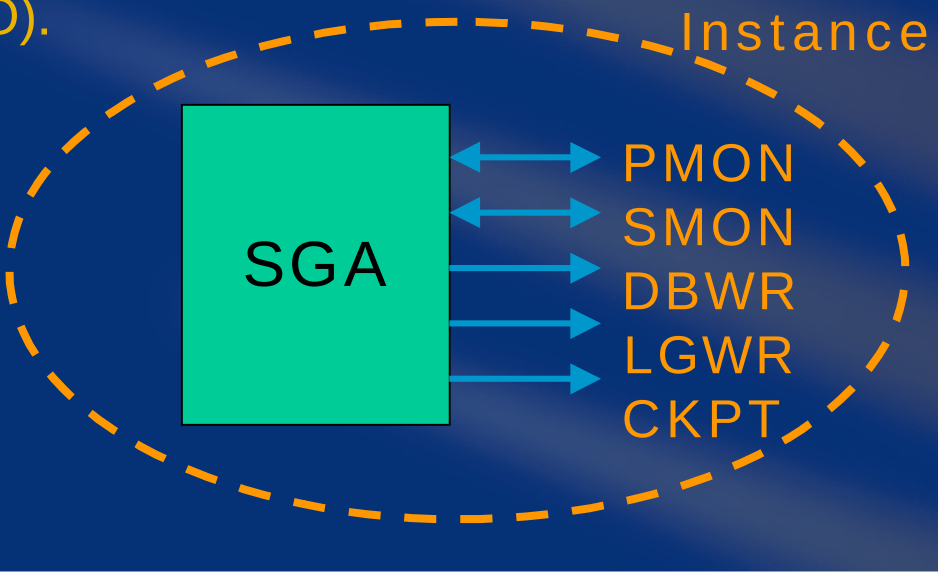
collective

Database Files

- **Data files:** holds actual user data. (e.g., tables, indexes, etc.)
- **Redo log files:** contains a record of all changes made to data residing in data files. **Should be multiplexed.**
- **Control file:** holds important Oracle metadata that is critical to its operation. **Should be multiplexed.**

Oracle Instance

- Oracle consists of a ***database*** and an ***instance***.
  - An ***instance*** is a combination of the pool of physical memory (RAM) allocated to Oracle, referred to as the **System Global Area (SGA)**, and the background processes that Oracle spawns to use this memory pool.
    - **SGA:** Area where Oracle attempts to cache database data for faster access (RAM I/O is about 1000x faster than disk I/O).

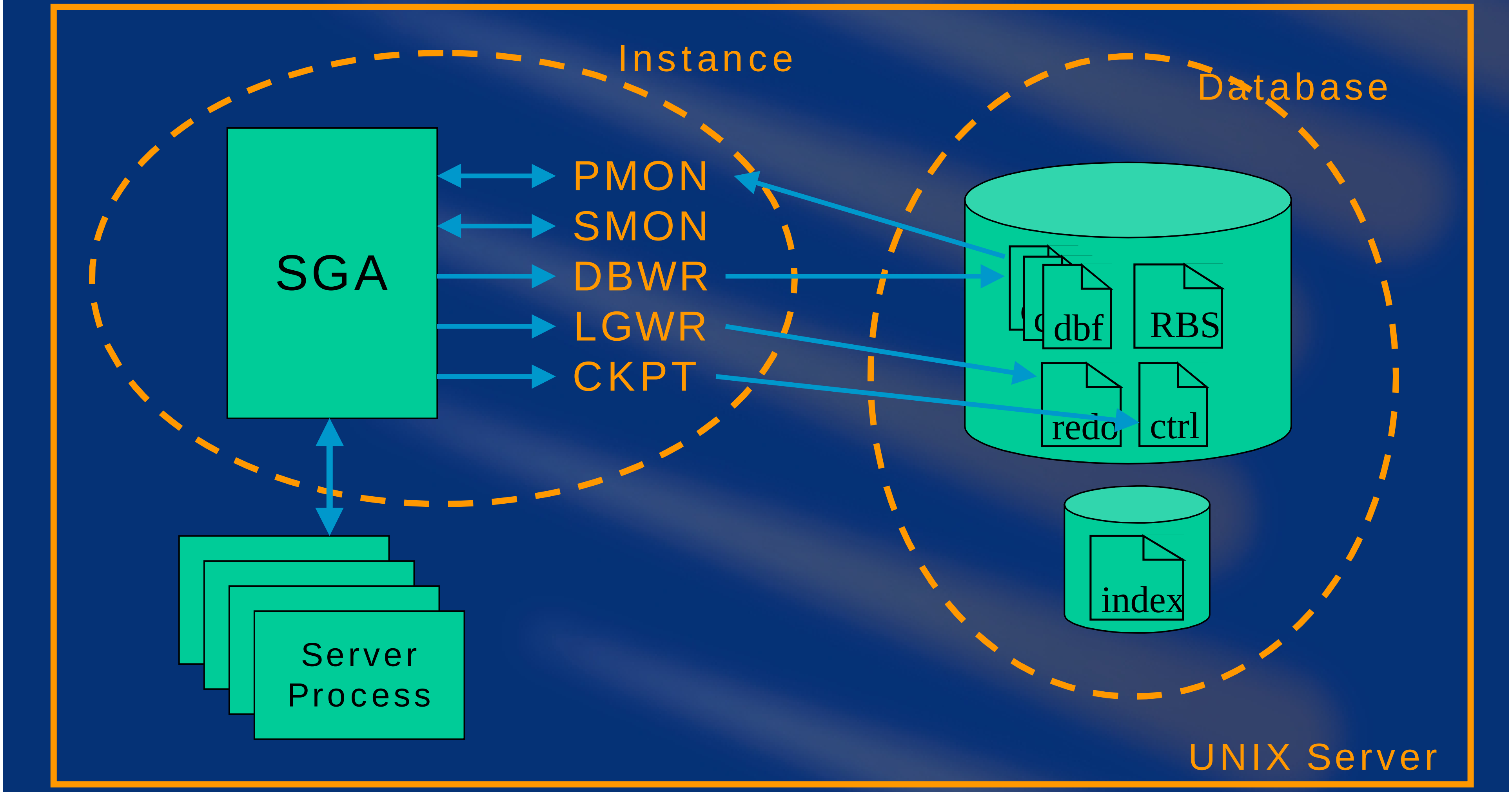


collective

Oracle Processes

- **DBWR:** process responsible for writing modified data that resides in the SGA to the data files on disk.
- **LGWR:** records changes applied to data in the redo log files.
- **PMON:** performs cleanup of failed or killed user and server processes.
- **SMON:** performs instance recovery should that database shutdown improperly.
- **CKPT: t**akes account of whenever DBWR writes data in memory to disk.

Oracle Server



Control File

- Contains all information necessary for an instance to access a database, during startup and normal operation.
- Metadata contained within it is important during database recovery, as it can identify files needed to bring a database to a stable condition.
- Due to its critical nature, should be multiplexed by Oracle and, optionally, mirrored by the OS.

collective

Data File

- **Data tables:** holds user data.
- **Indexes:** similar to book indexes -- contain table location information for faster lookups.
- **Rollback segments:** holds ***before-image*** copies of data being changed.
  - Maintains read consistency

Oracle Caching

- Whenever a user accesses a data table, an Oracle server process created on his behalf reads the data from the data file into the SGA.
- When another user attempts to access that same data, his server process will read from the copy in memory.
- If another user attempts to modify it, his server process will make changes to the copy in memory. **DBWR** will write these changes to disk in its due time.

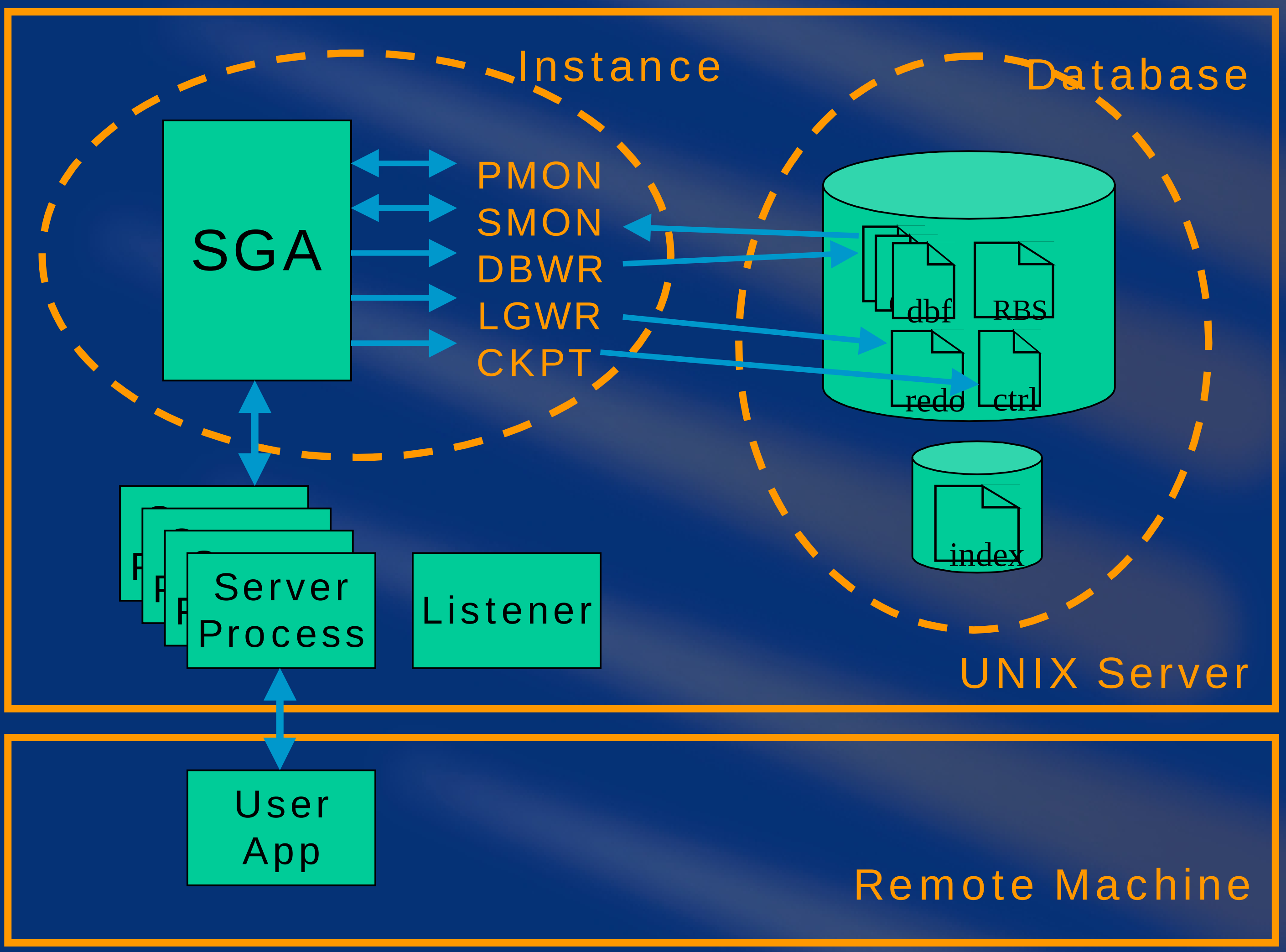
How can Oracle guarantee data consistency with DBWR’s write delay?

Redo Log Files

- Used to record changes made to data.
- Server processes started on a user's behalf will make changes on data in memory.
- A record of those changes are immediately recorded in the redo log files by **LGWR.**
- Should database crash before **DBWR** flushes changed data from memory to disk, on startup **SMON** will simply ***replay*** the redo log file to bring the database to a consistent state.
- Due to critical nature of redo log files, should be multiplexed and, optionally, OS mirrored.

collective

Oracle Environment



Unix

- Process Environment
- Virtual Memory
- Interprocess Communication

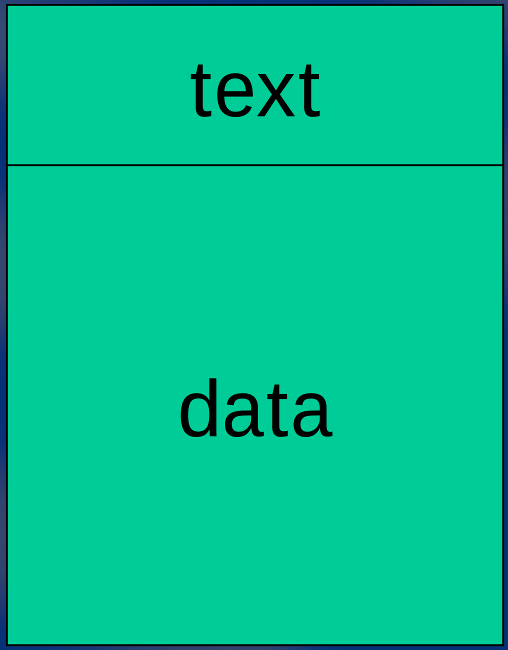
collective

Unix Process Environment

- Unix is a multi-user, multitasking operating system.
- Each process is given its share of system resources necessary for execution.
  - **CPU:** each process gets a *slice* of time to run
  - **Memory:** each process is allotted a portion of memory for execution (more on this later)

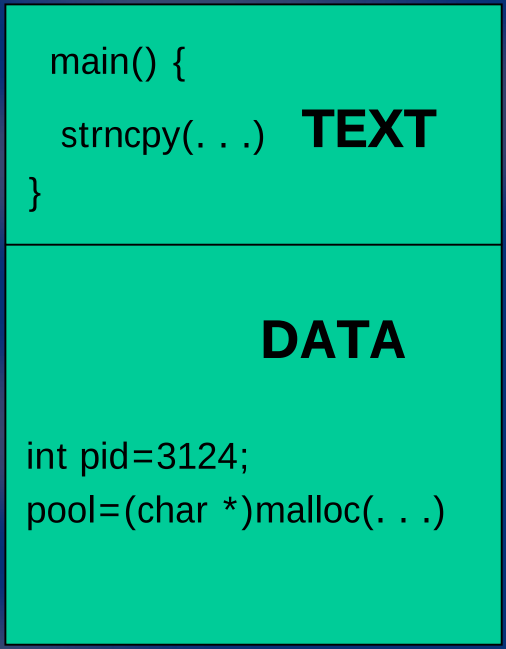
Process Address Space

- A process on a Unix system has, at a very high-level, an address space made up of a ***text*** and ***data*** portion.



Process Address Space Detailed

- **Text:** portion of address space where actual program instructions reside.
- **Data:** portion of address space where all data variables upon which the program's instructions operate.



TEXT

```
  main() {
   strncpy(. . .)
}
```

DATA

```
int pid=3124;
pool=(char *)malloc(. . .)
```

Execution Requirement

- A process **MUST** reside in memory for execution.
- Could cause problems on a system with limited RAM, but many processes running.
  - **How is this issue resolved?**

Virtual Memory

- **Definition**: Facility by which each process is given the illusion of having a large main memory at its disposal, although the computer may have relatively small memory.
- System uses secondary storage to store portions of a process's address space that does not fit in memory.
  - Commonly referred to as ***swap space*** or ***paging space.***

Paging

- Physical memory (RAM) is divided into page-sized chunks (typically 4k - 8k).
- Instead of moving an entire process's address space from memory to swap storage, page-sized granularity of displacement is performed.
  - This activity is referred to as ***paging***.
- Must allocate enough virtual memory to accommodate concurrently running processes.

Process Communication

How do processes with their own distinct address space communicate with one another?

Interprocess Communication

- **Definition**: facilities provided by Unix by which processes can communicate with one another.
- Commonly referred to as ***IPC.***

Common IPC Facilities

- **Signals:** serve primarily to notify a process of asynchronous events.
  - # kill -HUP 3214
  - generally about 31 signals available
- **Pipes:** unidirectional, first-in first-out, unstructured data stream of fixed max size.
  - Only used between related processes.

More IPC Facilities

- **Named pipes:** similar to regular pipes, except that they are persistent (maintain an entry in file system namespace)
  - Commonly referred to as ***FIFO.***
  - Can be used by unrelated processes.
- **Sockets:** communication endpoint that represents an abstract object on which a process can send/receive messages.
  - Commonly used in network communications.

collective

SysV IPC Facilities

- **Semaphores:** objects used to synchronize access to shared resources. Think of them as “locks.”
- **Message queues:** a header pointing to a linked list of messages. Each message contains a 32-bit “type” value, followed by the “data” area.
- **Shared memory:** a region of physical memory that is shared by multiple processes.
  - Singly the fastest method of IPC

Recap

- Oracle is multi-user, multi-processing database software.
- Unix provides a multi-user, multi-processing operating environment.
- What aspects of Unix are pertinent to a well-functioning Oracle database?

collective

Intermission

collective

Unix Considerations for Oracle

- Virtual Memory
- Unix IPC
- Physical Memory
- Disk Partitions

collective

Virtual Memory

- Typically, an area of disk used as backing store for memory-resident objects; used to present a virtual address space that is larger than the amount of RAM present.

Virtual Memory and Oracle

- Paging space is recommended to be 1.5x - 3x the amount of RAM on system.
  - A more accurate assessment can be made with tools like ‘ps’, ‘svmon’, ‘pmap’, ‘pmem’, etc.
- The address space requirements of all the numerous Oracle background and server processes require such large swap allocations.

Unix IPC

- Facilities provided by which processes can communicate with one another.
- Common IPC include signals, pipes, FIFOs, sockets, semaphores, message queues, and shared memory.

Unix IPC and Oracle

Two Unix IPC facilities are of utmost importance to Oracle:

- **Shared Memory**
- **Semaphores**

collective

Shared Memory

- Physical RAM pages that are shared among multiple processes.
- Oracle uses it to implement its **System Global Area (SGA)**, in which table data, RDBMS metadata, and other Oracle objects are cached.

Shared Memory and the SGA

- The larger the SGA, the better, as frequent I/O in RAM is significantly faster than the same from disk.
- SGA size must be balanced with memory requirements of the OS and other running applications.
  - Too much results in excessive ***paging.***
  - Too little results in high disk I/O.
- Unix variants offer unique shared memory features to enhance the performance of Oracle.

collective

Common Shared Memory Issues

- Unix Errors
  - ENOSPC: the kernel setting for the the number of shared memory segments globally is too low.
  - ENOMEM: not enough paging space allocated to accommodate the SGA’s size.
- Oracle Errors:
  - Too many segments needed.
    - Shared memory maximum is too small (shmmax)
    - Maximum number of shared memory segments per process is too small (shmseg)

Semaphores

- Kernel objects used as a means to synchronize access to shared resources.
- Oracle uses them as “latches”, or locks, to synchronize access among all the background and server processes to its shared resource, the **SGA.**

Semaphores and Oracle

- Oracle requires that there, at least, be as many semaphores as there are Oracle processes (server and background combined).
- Reason for this is unclear from Oracle.
  - Likely used for process-to-process communication, instead of signals.

Common Semaphore Issues

- Unix Errors:
  - – ENOSPC: not enough semaphore structures set to accommodate Oracle’s request.
- Oracle Errors:
  - – “post/wait driver initialization failed”
  - – Oracle couldn’t grab enough semaphores to accommodate estimated number of processes.
    - Oracle checks a variable named PROCESSES to calculate the number of semaphores needed.

Physical Memory (RAM)

- Due to the memory needs of Oracle's server, background processes, and its SGA, a large amount of RAM is highly desirable.
  - Entire SGA **must** fit in RAM.
- When drafting specifications for server hardware of a Unix system that will host Oracle, be sure to invest in plenty of RAM.

Disk Partitions

- Needed for allocation of data files, redo log files, and control files.
- Without disk partitions, not much you can do with Oracle. ☺

collective

Raw vs. Cooked

- Raw disk refers to a disk slice containing no file system.
- Cooked disk refers to a disk that is formatted with a file system.
- Common file system types on Unix are ufs, xfs, jfs, vxfs, advfs, etc.

collective

Historical Benefits of Raw Disk

- Raw disks were commonly chosen for performance due to the fact that disk I/O to such devices didn’t have to go through file system code and buffers to access the disk.
- Historically, file systems added too much overhead for Oracle’s I/O characteristics
  - File-system buffer

collective

Modern Advantages of Cooked Disk

- Advances in file system technology have narrowed that gap significantly in I/O performance.
  - Asynchronous I/O
  - Direct I/O
  - Configurable file-system buffers
  - Veritas File-systems Quick I/O
  - extent-based file systems (xfs, jfs2, vxfs, etc.)
- File-systems provide more flexibility.
  - ‘cp’, ‘mv’, ‘cpio’, ‘tar’, ‘dump/restore’, etc.

collective

Which Should You Use?

- If using Oracle Parallel Server (OPS), you **must** use raw disks.
- Otherwise, use a modern file-system.
  - This is only the opinion of the presenter (and a bunch of expert Oracle book authors). ☺

collective

RAID and Oracle

- **RAID 0: Striping**
  - Just pure disk striping.
  - Great for Oracle disk I/O performance.
  - However, offers no additional protection.

- **RAID 1: Mirroring**
  - Offers data redundancy.
  - Great for data reliability which is essential for Oracle, and great for read performance.
  - However, adds some write overhead, and is an expensive solution.

RAID 5 vs RAID 10

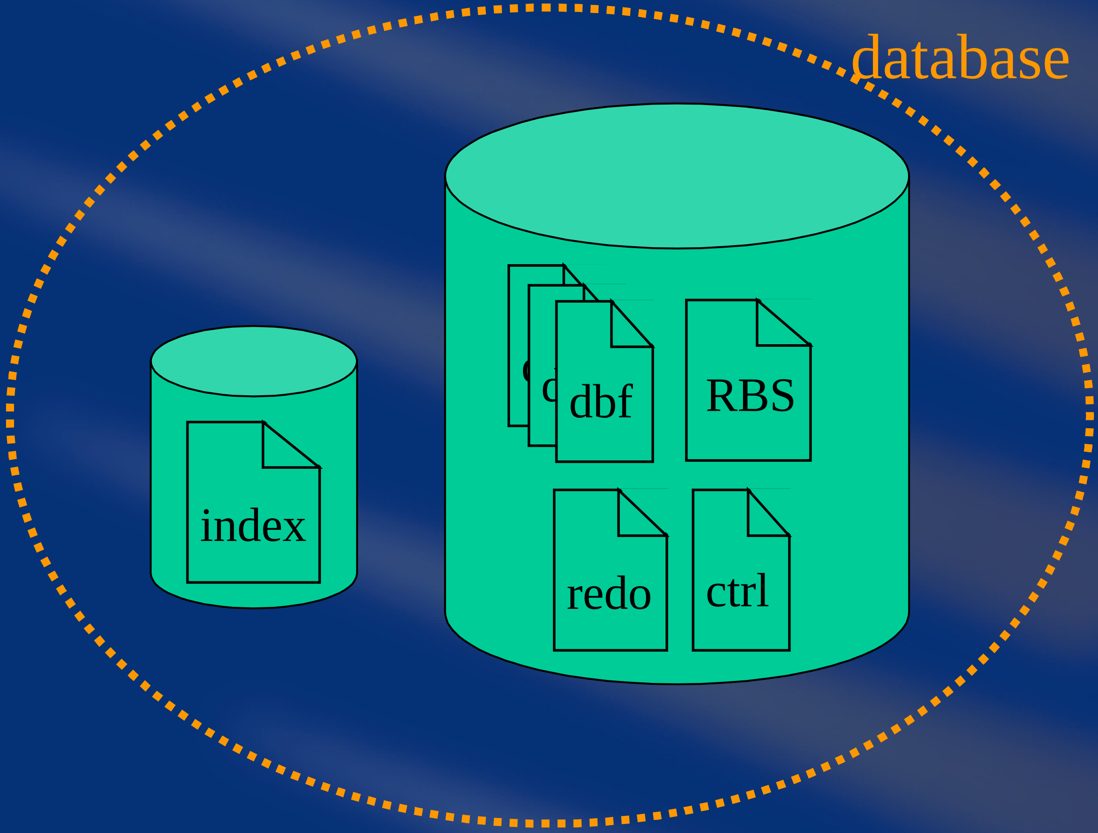
- **RAID 5: Data & Parity Striping**
  - Data & Parity striped across all disks.
  - Inexpensive alternative to RAID 0, and improves read performance.
  - Degrades write performance due to parity calculations for each write. **DBAs like to request this!!!!**

- **RAID 10: Striping and Mirroring**
  - Disks that are both striped and mirrored.
  - Best of both worlds (reliability and performance).
  - Expensive solution, for reasons stated for RAID 1.

collective

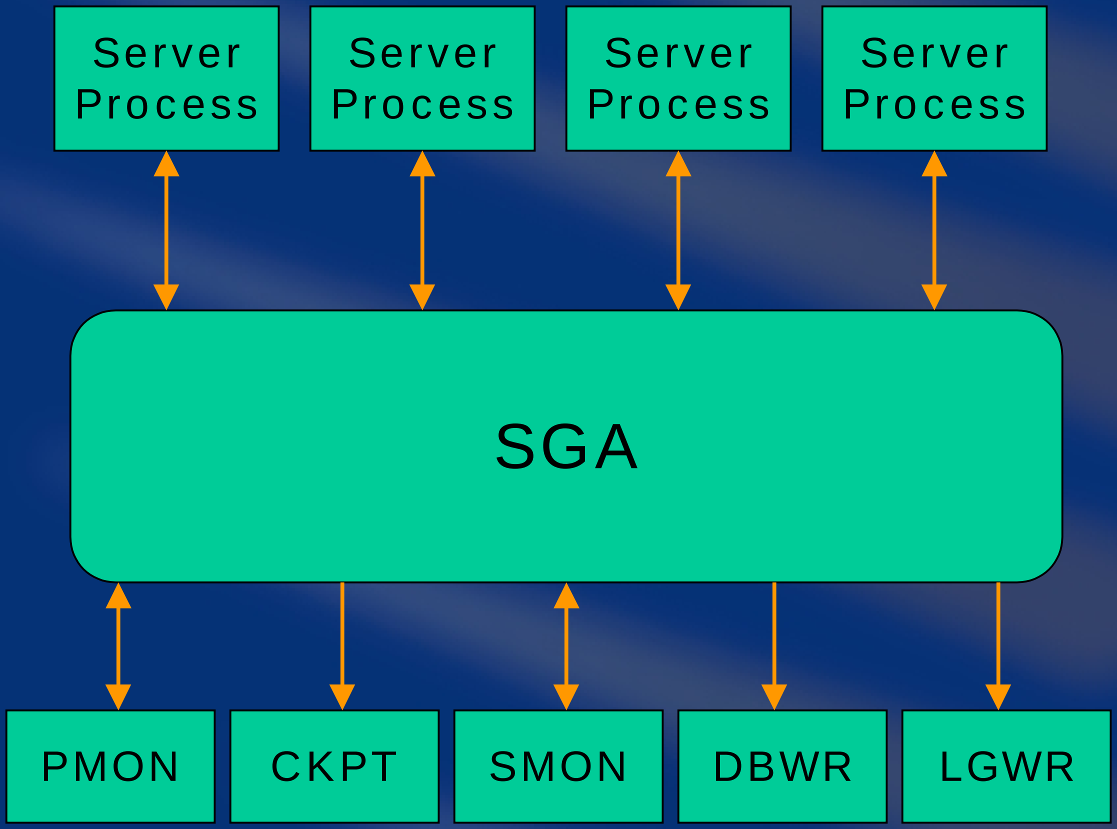
Summary

An Oracle Database



collective

An Instance



Conclusion

- Oracle's multi-user, multi-processing nature meshes well with the powerful multi-user, multi-processing Unix OS.
- An understanding of how the two interact leads to better architectures and support of these often combined technologies.
- To meet the demands of our customer SLAs, it behooves us to better understand entire environment.

Major Thanks To

David J. Young

Matt Coffey

Illinois District of Collective Technologies

collective

Questions?

E-mail: medawson@colltech.com

Slides: http://www.uniforum.chi.il.us/slides/oracle