



# Introduction to Oracle for Sybase Users

Stephen Darlington  
10th October, 2002.



# Introduction

- What I'm going to cover
- What I'm *not* going to cover
- What do I know?



## What's in the Box?

- Database (obviously)
- Jserver (Oracle 8i JVM)
- Java stored procedures and triggers
- Java application server
- JSP / Servlets
- J2EE Container
- Java -> C compiler



## What's In The Box? (2)

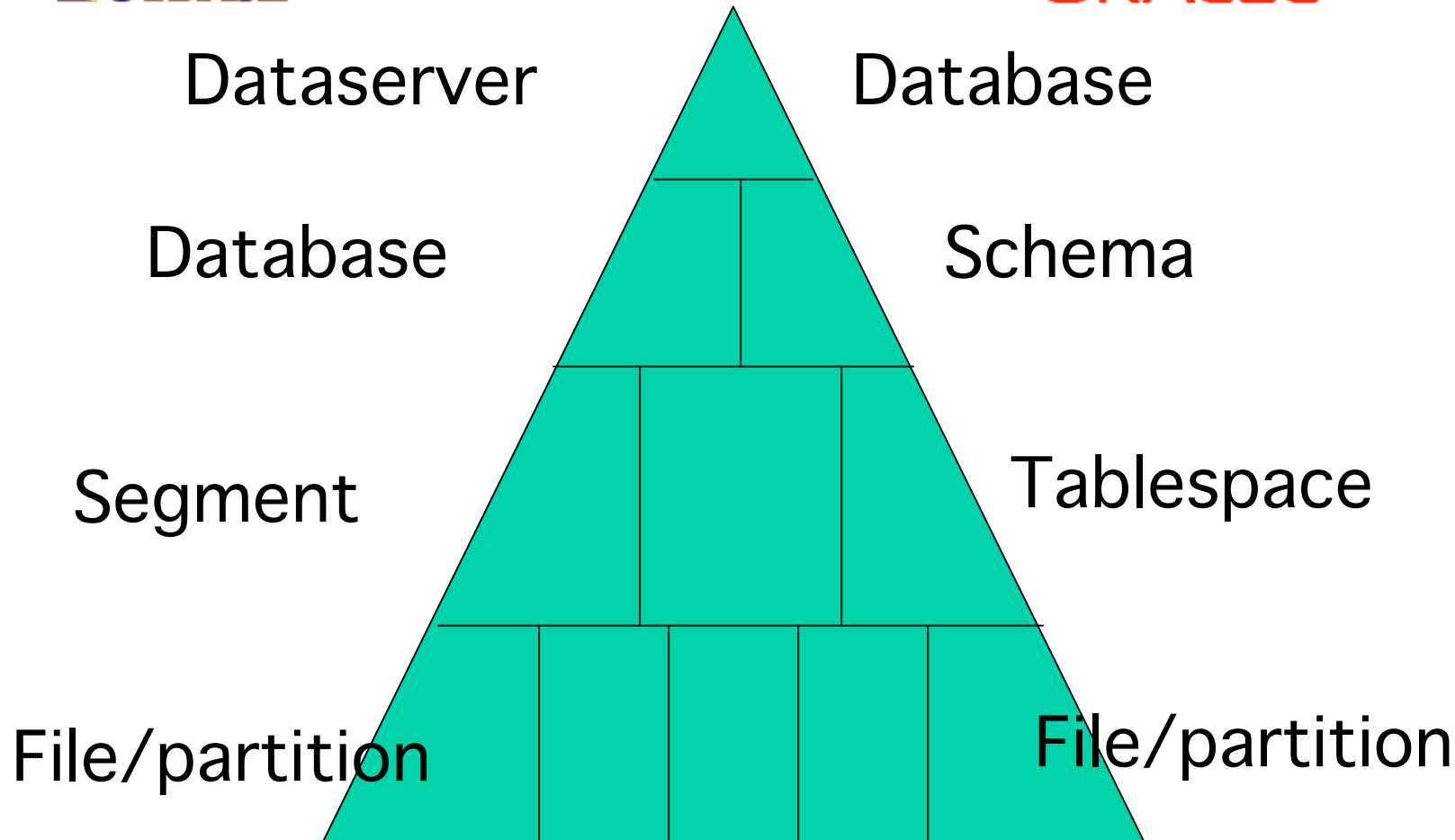
- Oracle HTTP Server
- Oracle Enterprise Manager
- Oracle Internet Directory
- Oracle internet Filesystem (iFS)
- Advanced Queuing
- Precompilers
- Monitoring tools, parallel server, clustering, XML...



# Architecture Summary

 SYBASE

ORACLE





# Oracle Users

- Very different from Sybase!
- Schema and user the same thing...
- ...but an Oracle user is more like a Sybase database
- Synonyms
- Object search path



# Oracle SQL

- SQL\*Plus
- Generally considered to be more powerful than Sybase...
- ...but is non-standard in places
- Useful extensions



# Oracle PL/SQL

- Full “proper” language modelled on Ada
- (Not as bad as it sounds!)
- Has packages, functions, procedures and pseudo-OO bits
- SQL built into the language...
- ...but implemented separately to the SQL engine
- Client and server versions





## PL/SQL Example 1: Convert GetTraderDetails

```
create procedure GetTraderDetails (  
    @trader char(15)  
) as  
begin  
    select  
        TRADER = @trader,  
        TRADERLOC = traderloc,  
        PRINTER = printer,  
        HOME = home,  
        SUBSID = subsid,  
        TOUPPER = toupper,  
        PURPOSE = purpose,  
        STRATEGY = strategy,  
        CALENDAR = calendar,  
        CURR = curr  
    from  
        Trader  
    where  
        trader = @trader  
end
```



# PL/SQL Example 1: Converted

```
CREATE VIEW traderdetails AS
  SELECT
    trader,
    traderloc,
    printer,
    home,
    subsid,
    toupper,
    purpose,
    strategy,
    calendar,
    curr
  FROM
    trader
```

Transact-SQL and PL/SQL are not  
directly equivalent!



## PL/SQL Example 2 (Package)

```
CREATE OR REPLACE PACKAGE app_employee AS

    FUNCTION checkEmp (name IN employee.name%TYPE)
        RETURN BOOLEAN;

    PROCEDURE addEmp (newrec IN employee%ROWTYPE);

    no_employee EXCEPTION;
    many_employee EXCEPTION;

END;
```



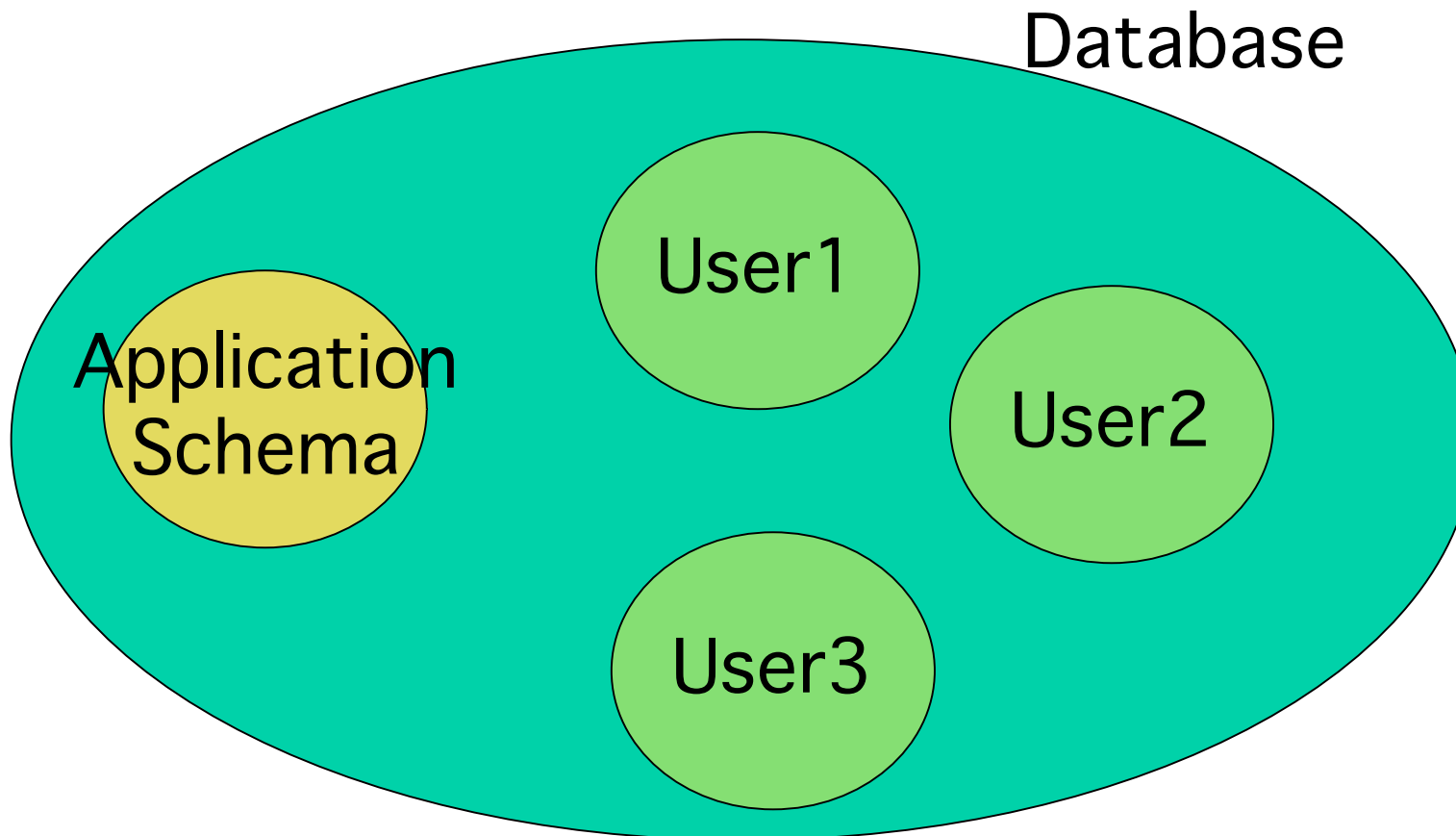
## PL/SQL Example 2 (Package Body)

```
CREATE OR REPLACE PACKAGE BODY app_employee AS
    FUNCTION checkEmp (name IN employee.name%TYPE)
        RETURN BOOLEAN
    IS
        retval BOOLEAN;
        salary INTEGER;
    BEGIN
        SELECT e.salary
        INTO salary
        FROM employee e
        WHERE e.name like name || '%';
        RETURN (salary > 0);
    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            RAISE no_employee;
        WHEN DUP_VAL_ON_INDEX THEN
            RAISE many_employee;
    END;
    PROCEDURE addEmp (newrec IN employee%ROWTYPE)
    IS
    BEGIN
        NULL;
    END;
```

END;



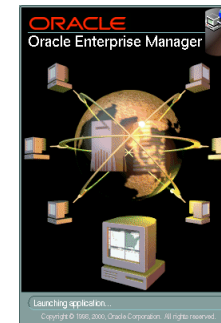
# PL/SQL Application





## If You're A DBA...

- SQL\*Plus is your friend (really)
- May also hear of “sysmgri”
- Imp/exp
- Oracle Enterprise Manager
- Lot's of third party tools, including TOAD and SQL Navigator





## Other Cool Stuff

- Materialised views
- Function based indexes
- Partitioned tables
- File I/O in PL/SQL
- XML parser in Java, PL/SQL, C++



## In Conclusion...

- Way too much to cover in any detail
- Basics no more complicated than Sybase
- ...but there is much more to it.