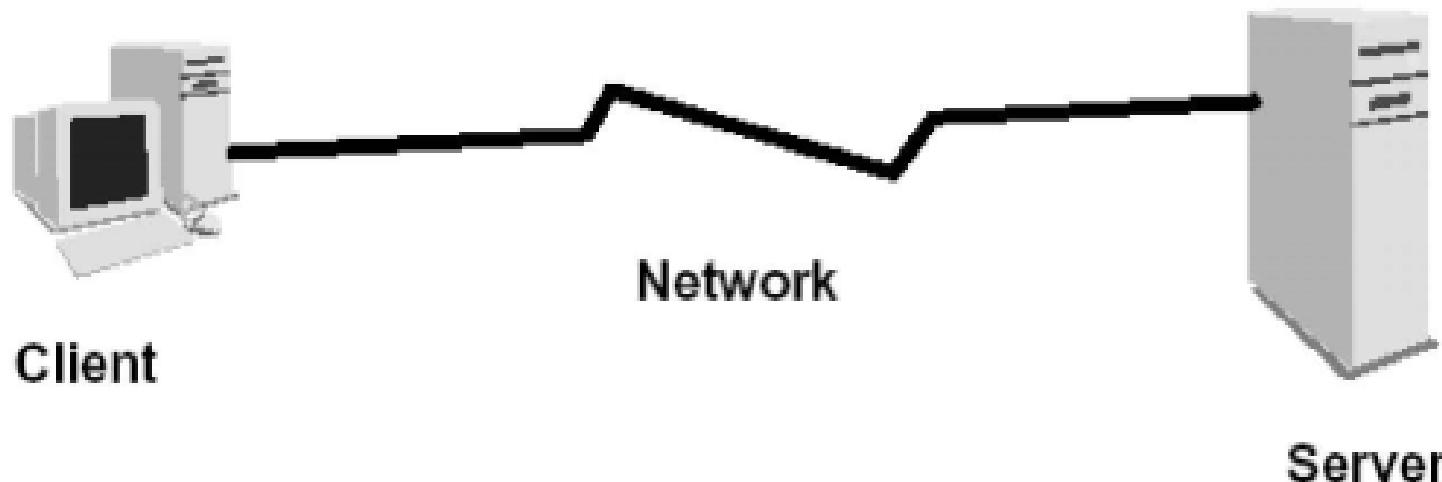


# Network Environment Challenges

- Configuring the network environment
- Maintaining the network
- Tuning, troubleshooting, and monitoring the network
- Implementing security in the network
- Integrating legacy systems

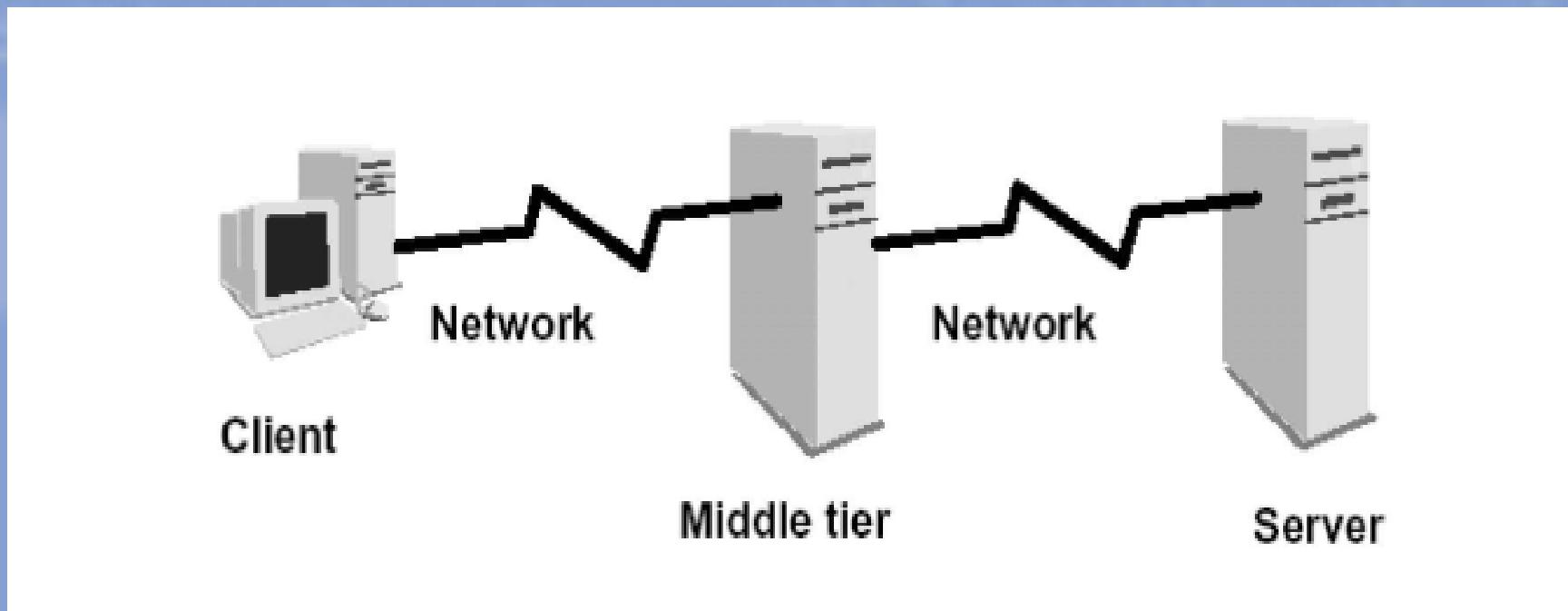
# Simple Network: Two-Tier

- Network connects client and server
- Client and server speak the same “language” or protocol

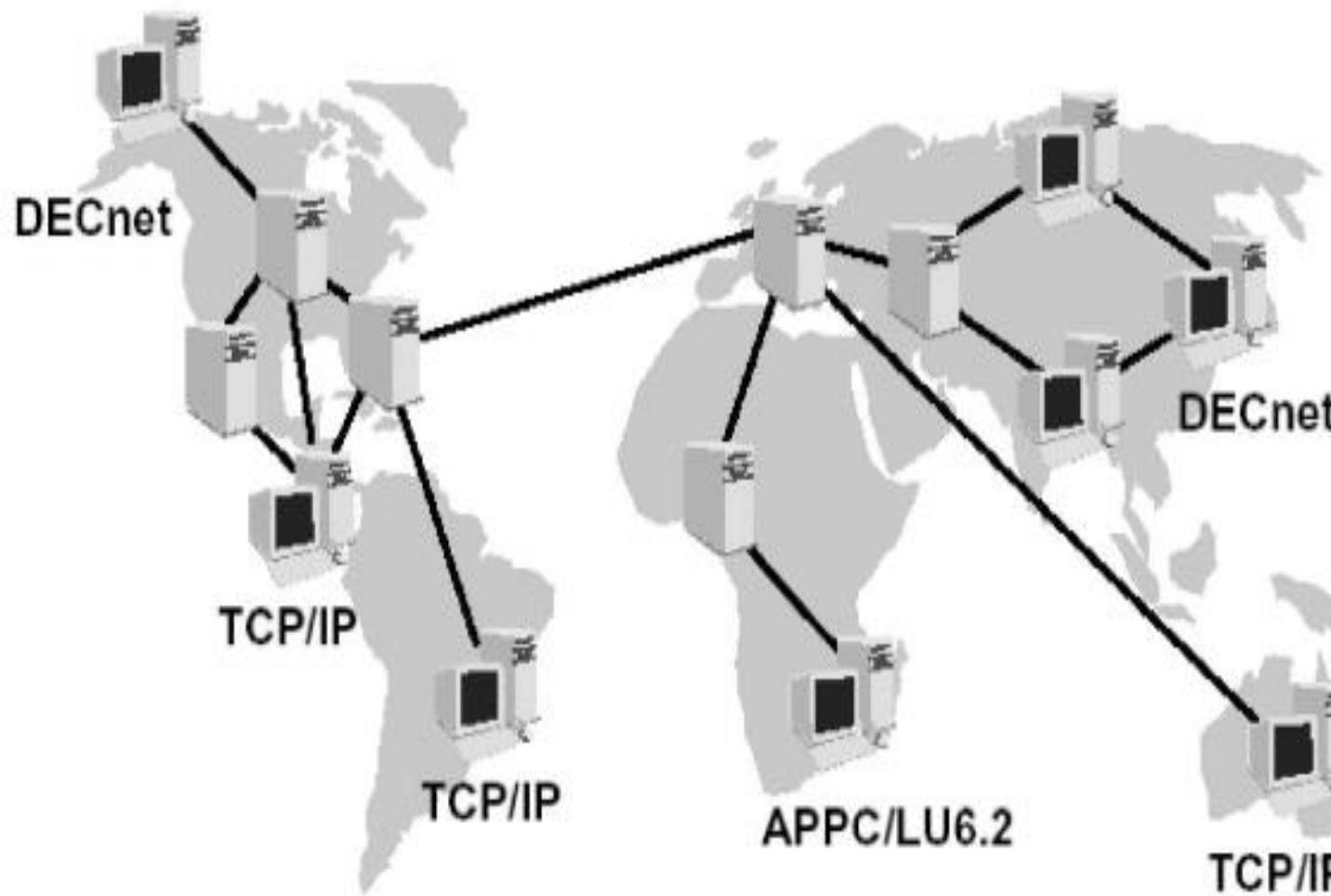


# Simple to Complex Network: N-Tier

- Client can be a thin client or a PC
- Middle tier can contain applications and services
- Server holds actual data



# Complex Network



# Oracle Networking Solutions

- Connectivity
- Directory Services
- Security
- Accessibility



# Connectivity: Oracle Net Services

- Protocol independence
- Comprehensive platform support
- Integrated GUI administration tools
- Multiple configuration options
- Tracing and diagnostic toolset
- Basic security

# Security: Oracle Net and Firewalls

- Oracle works with key firewall vendors to provide firewall support
- Oracle Net Application Proxy Kit allows firewall vendors to provide connection support for Oracle environment
- Oracle Net Application Proxy is based on Connection Manager
- Oracle supports two categories of firewalls:
  - Proxy based firewalls
  - Statefull packet inspection firewalls

# Accessibility : Heterogeneous Services

- Enables access of legacy data as if it resides in a single ,local relational database
- Enable Oracle procedure calls to access non-Oracle systems, services, or APIs

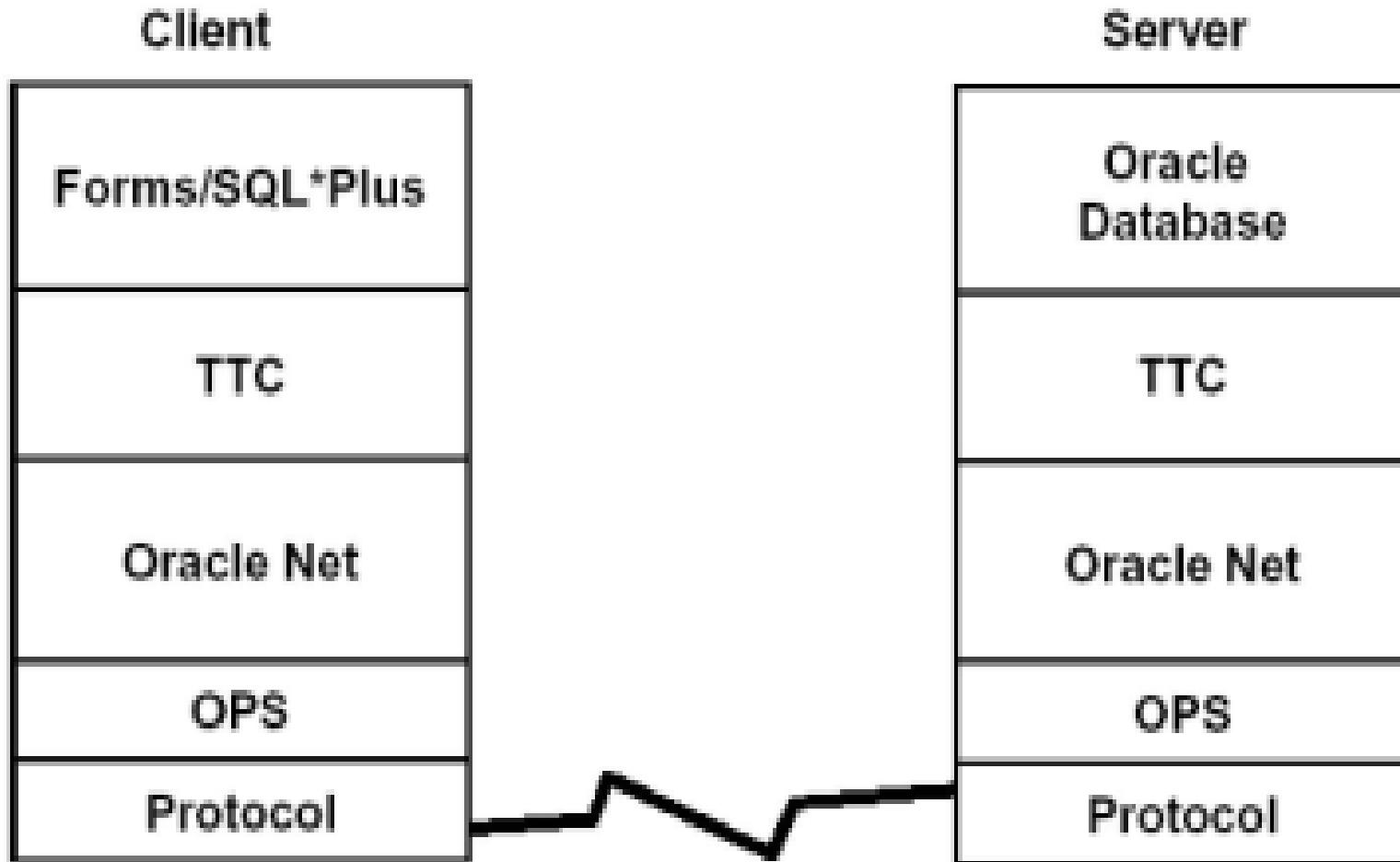
# Accessibility: External Procedures

- External procedures are functions written in a 3GL language that can be called from PL/SQL
  - Support of external procedures allows the developer more flexibility than SQL or PL/SQL provide
- The Oracle listener can listen for external procedure calls
- Connections to external procedure can be configured during or after server installation

# Oracle Net Connections

- Oracle Net is used to establish connections between applications on a network depending on the following
  - The network configuration
  - The location of the nodes
  - The application
  - The network protocol
- The connections types can be
  - Client-Server Application
  - Web Application Connection

# Client-Server Application Connection: No Middle-Tier

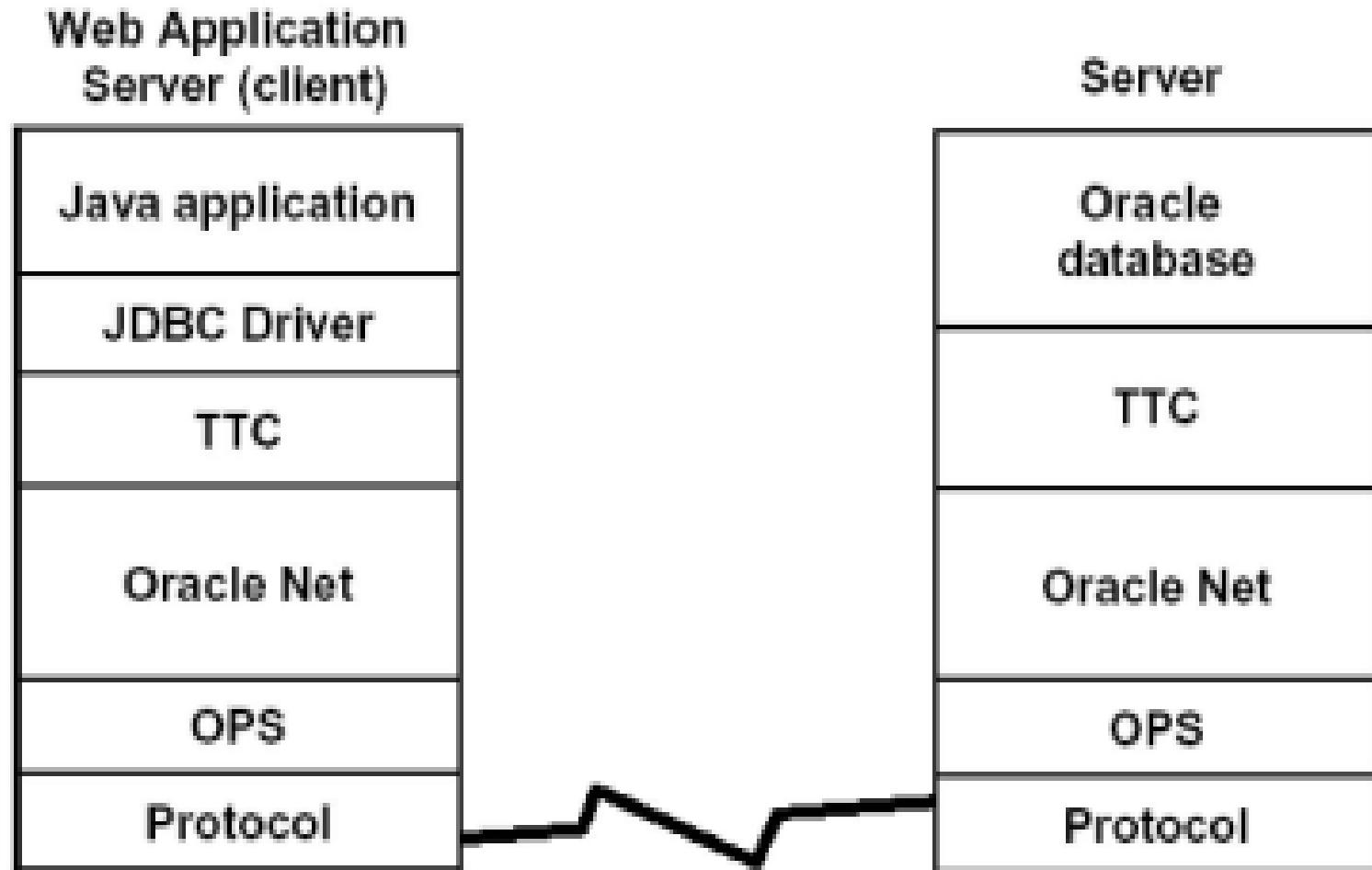


# Web Client Application Connections

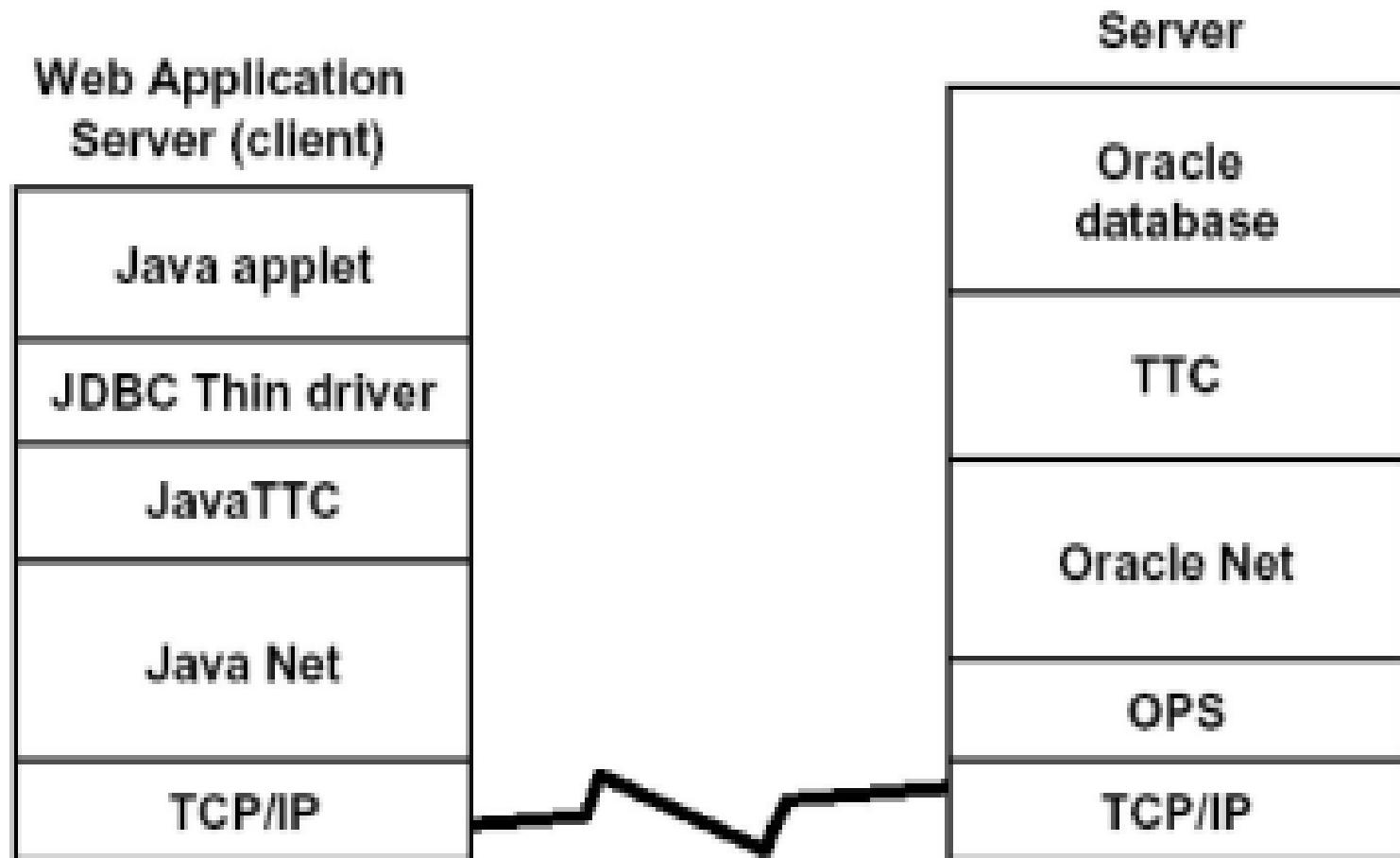
**Web browsers can connect to an Oracle server in the following ways:**

- Using a Web Server as a middle tier configured with a
  - JDBC Oracle Call Interface (OCI) driver
  - Thin JDBC driver
- Connecting directly to an Oracle Server using
  - IIOP
  - HTTP

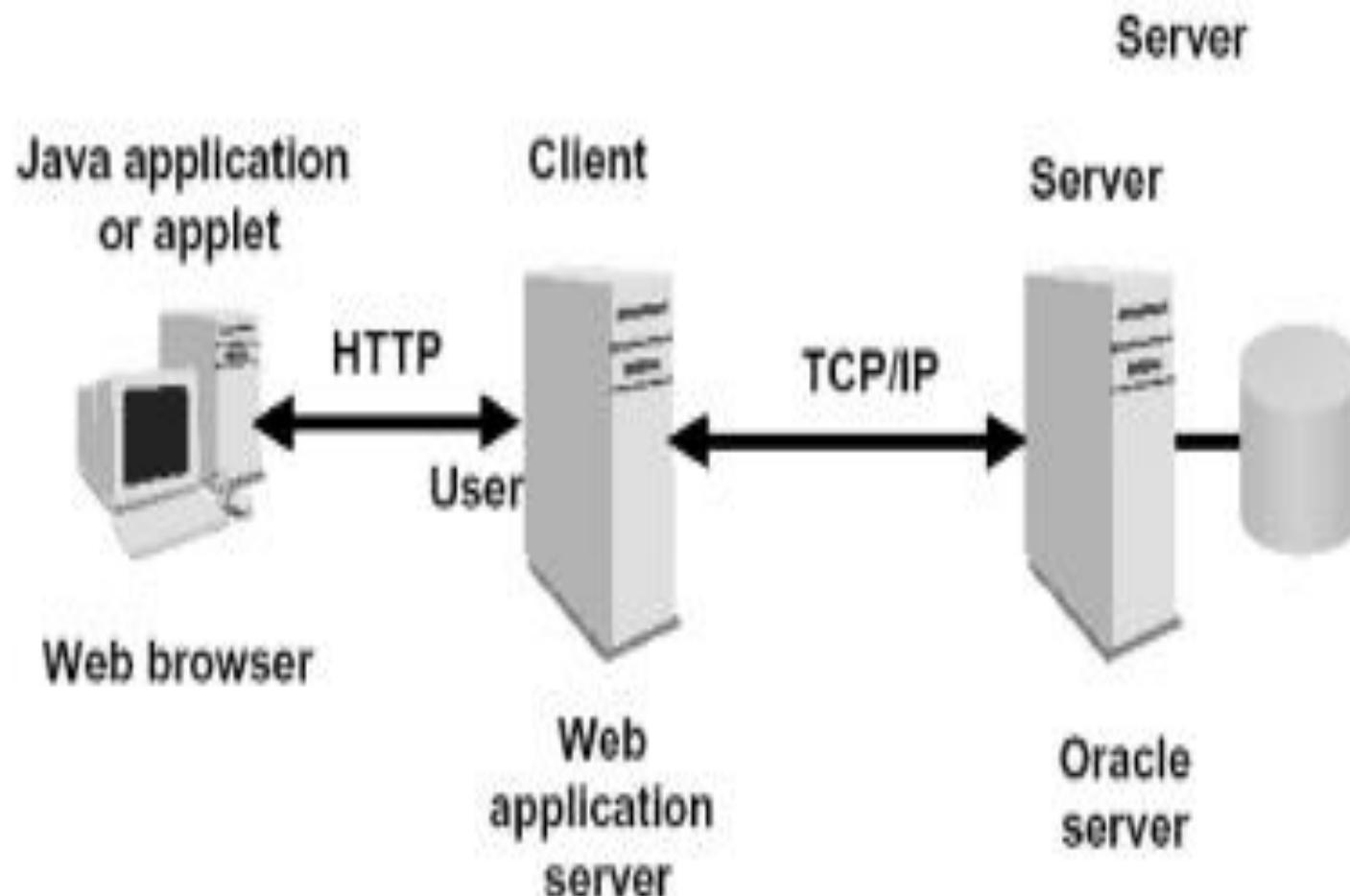
# Client-Server Application Connection: Java Application Client



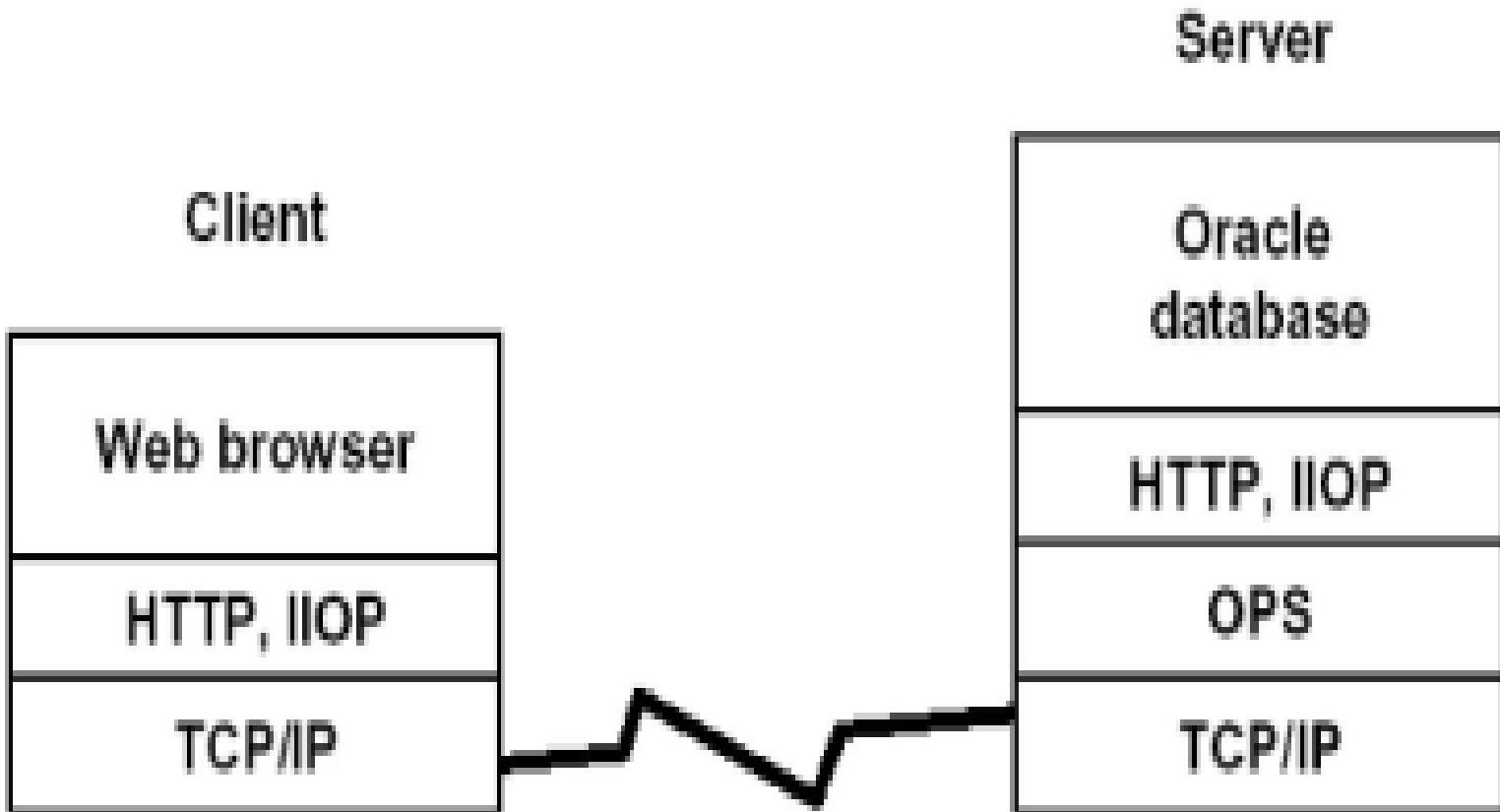
# Web Client Application Connection: Java Application Client



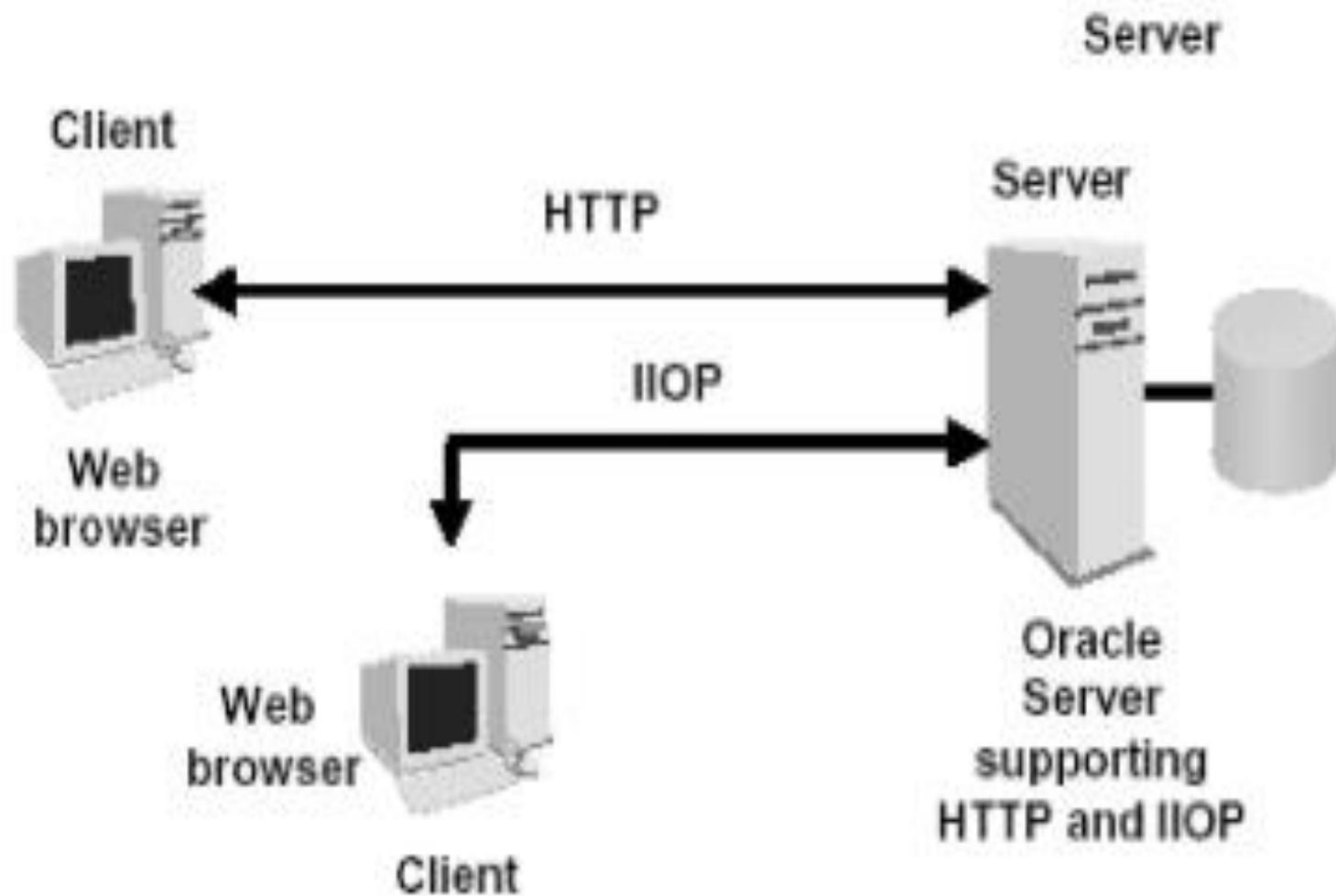
## Web Client Application Connection : Web Server Middle - Tier



# Web Client Application Connections : No Middle - Tier



# Web Client Application Connections : No Middle - Tier



# Data Dictionary

- During database creation, the Oracle server creates additional object structures within the data files
  - Data dictionary tables
  - Dynamic performance tables

# Data Dictionary

- The data dictionary is a set of read-only tables and views that record, verify, and provide information about its associated database.
- Describes the database and its objects
- Includes two types of objects
  - Base tables
    - Store description of database
    - Created with CREATE DATABASE
  - Data Dictionary views
    - Summarize base table information
    - Created using catalog.sql script

# Data Dictionary Contents

The data dictionary provides information about:

- Logical and physical database structure
- Definitions and space allocations of objects
- Integrity constraints
- Users
- Roles
- Privileges
- Auditing

# How the Data Dictionary Is Used

- The data dictionary has three primary uses:
  - The Oracle server uses it to find information about:
    - Users
    - Schema objects
    - Storage structures
  - The Oracle server modifies it when a DDL statement is executed
  - Users and DBAs can use it as a read-only reference for information about the database

# Data Dictionary View Categories

The data dictionary consists of three main sets of static views distinguished from each other by their scope:

- DBA: What is in all the schema
- ALL: What the user can access
- USER: What is in the user's schema

**DBA\_xxx**

All of the objects in the database

**ALL\_xxx**

Objects accessible by the current user

**USER\_xxx**

Objects owned by the current user

# Dynamic Performance Tables

- Dynamic performance views record current database activity.
- Views are continually updated while the database is operational
- Information is accessed from:
- Memory
- Control file
- DBA uses dynamic views to monitor and tune the database
- Dynamic views are owned by SYS user
- DML is not allowed

## Querying the Data Dictionary and Dynamic Performance Views

- Data dictionary and dynamic performance views can be queried for information.
- A listing of views available can be retrieved by querying the DICTIONARY view.
- A listing of the columns and its contexts can be accessed using DESCRIBE and SELECT.
- Column comments are available to retrieve more insight into what a column context means within a particular view.

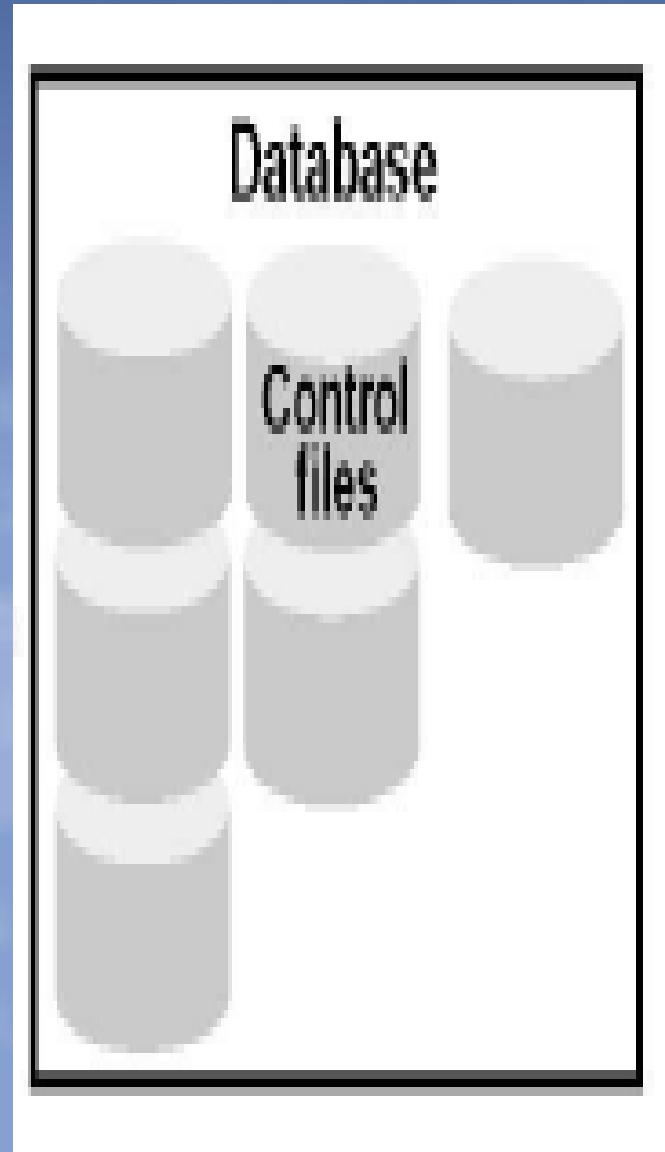
# Data Dictionary Examples

- General Overview
  - DICTIONARY, DICT\_COLUMNS
- Schema Objects
  - DBA\_TABLES,DBA\_INDEX,DBA\_TAB\_COLUMNS,  
DBA\_CONSTRAINTS
- Space Allocation
  - DBA\_SEGMENTS,DBA\_EXTENTS
- Database Structure
  - DBA\_TABLESPACES, DBA\_DATA\_FILE

# Control File

The control file is a binary file that defines the current state of the physical database...

- Loss of the control file requires recovery
- Is required to operate
- Is linked to a single database
- Should be multiplexed
- Maintains integrity of database
- Sized initially by CREATE DATABASE



# Control File Contents

- A control file contains the following entries:
- Database name and identifier
- Time stamp of database creation
- Tablespace names
- Names and locations of data files and redo log files
- Current redo log file sequence number
- Checkpoint information
- Begin and end of undo segments
- Redo log archive information
- Backup information

# Multiplexing the Control File - Using SPFILE

- Alter the SPFILE

```
ALTER SYSTEM SET control files =
'$HOME/ORADATA/u01/ctrl01.ctl',
'$HOME/ORADATA/u02/ctrl02.ctl' SCOPE=SPFILE;
```

- Shutdown normal

```
SQL> shutdown normal
```

- Create additional control files

```
$ cp $HOME/ORADATA/u01/ctrl01.ctl
$HOME/ORADATA/u02/ctrl02.ctl
```

- Start the database

```
SQL> startup
```

# Multiplexing the Control File - Using init.ora

- Shut down the database in normal status

```
SQL> shutdown normal;
```

- Copy the existing control file to a new name and location

```
$ cp /DISK1/ctrl01.ctl /DISK2/ctrl02.ctl
```

- Add the new control file name to init.ora

```
CONTROL_FILES = (/DISK1/ctrl01.ctl,  
/DISK2/u02/ctrl02.ctl)
```

- Start the database

```
SQL> startup
```

# Managing Control Files with OMF

- Control files are OMF CREATED if the CONTROL\_FILES parameter is not specified.
- OMF control files are located at DB\_CREATE\_ONLINE\_LOG\_DEST\_N
- Control file names are uniquely generated and displayed in the alertSID.log file when files are created.

# Obtaining Control File Information

- Information about control file status and locations can be retrieved by querying the data dictionary
- V\$CONTROLFILE: List the status of all control files associated with the instance
- V\$PARAMETER: List status and location of all parameters
- V\$CONTROLFILE\_RECORD\_SECTION: Provides information about the control file record sections
- SHOW PARAMETERS CONTROL\_FILES: List the name, status, and location of the control files