USIU AFRICA

United States International University-Africa

DATABASE SYSTEMS

Presented by Prof. Elisha T. O. Omulo

USIU AFRICA

WEEK 11 AGENDA

- **The Native SQL connectivity.**
- **The Microsoft's Open Database Connectivity (ODBC).**
- **The Java Database Connectivity (JDBC)**
- **The Database Internet Connectivity.**

Course Textbook: **Carlos Coronel, Steven Morris, Peter Rob and Keeley Crockett Database Principles: Fundamentals of Design, Implementation, and Management, 14th Edition, 2022, ISBN-13978-0357673034.**

Database connectivity tools (ODBC, JDBC)

Database connectivity- is the mechanisms through which application programs connect and communicate with data repositories.

- **Middleware- Database connectivity software providing an interface between the application program and the database.**
- **Data source - The data repository, in the data management application, such as Oracle RDBMS, SQL Server DBMS, or IBM DBMS, that is used to store the data generated by the application program.**
- **Data source or data repository may be located anywhere and hold any type of data for various databases eg a relational database, a NoSQL database, a spreadsheet, or a text data file.**

Task: Discuss database connectivity ideas.

Database connectivity tools (ODBC, JDBC)

Database connectivity- is the mechanisms through which application programs connect and communicate with data repositories.

- **There are many ways to achieve database connectivity, but we consider:**
- **Native SQL connectivity (vendor provided)**
- **Microsoft's Open Database Connectivity (ODBC),**
- **Java Database Connectivity (JDBC)**

Task: Discuss database connectivity options.

Database connectivity tools (ODBC, JDBC)

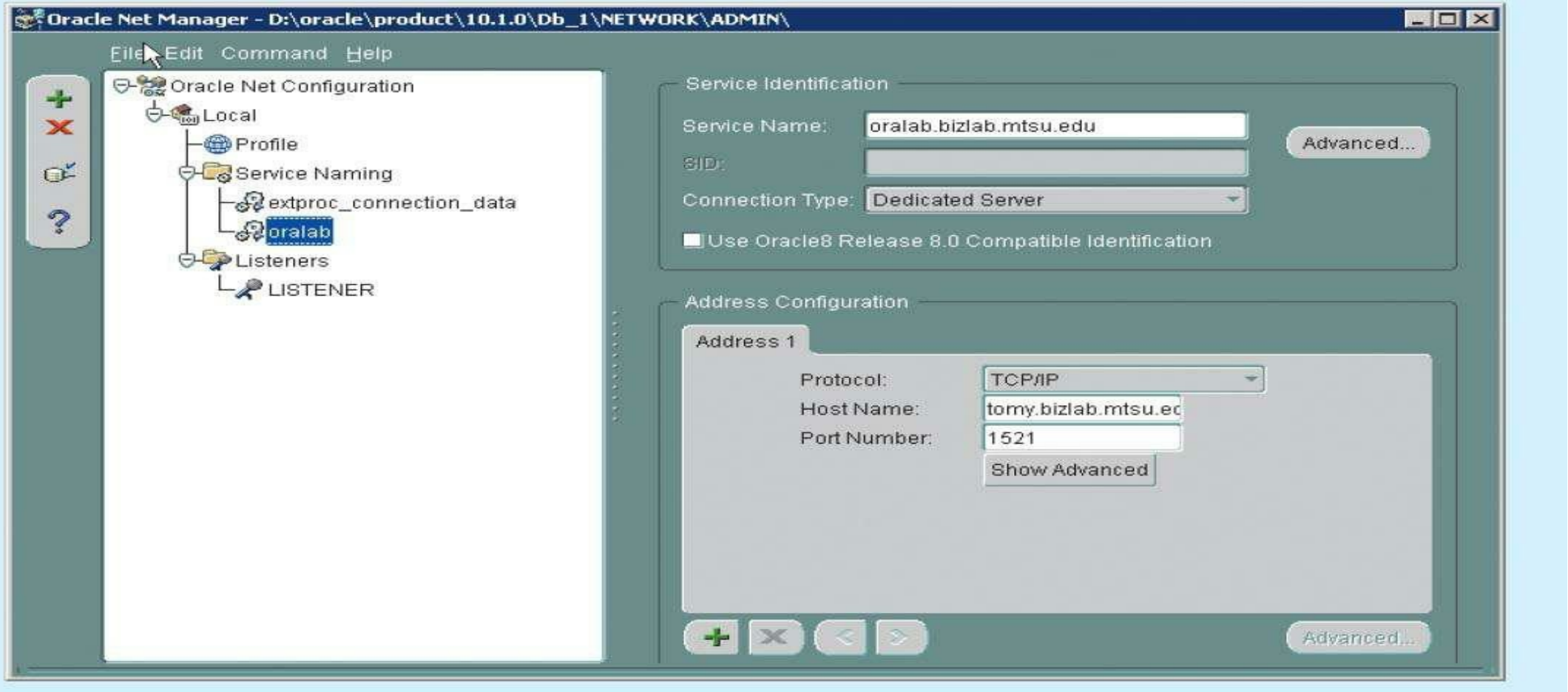
NATIVE SQL CONNECTIVITY

- Native SQL connectivity refers to the connection interface that is provided by the database vendor and is unique to that vendor eg. Oracle RDBMS.
- To connect a client application to an Oracle database, you must install and configure Oracle's SQL*Net interface in the client computer.
- Native database connectivity interfaces are optimized for "their" DBMS, and those interfaces support access to most or all of the database features.
- However, maintaining multiple native interfaces for different databases can become a burden for the programmer.
- Most current DBMS products support other database connectivity standards, the most common being ODBC.

Task: Discuss database native connectivity.

Database connectivity tools (ODBC, JDBC)

NATIVE SQL CONNECTIVITY



Task: **Discuss oracle native database connectivity.**

Database connectivity tools (ODBC, JDBC)

Open Database Connectivity(ODBC)

- **ODBC was developed by Microsoft's in the 1990s.**
- **It implemented a superset of the SQL Access Group Call Level Interface (CLI) standard for database access.**
- **It is one of the most widely supported database connectivity interface.**
- **ODBC allows any Windows application to access relational data sources, using SQL via a standard application programming interface (API).**
- **An API is "a set of routines, protocols, and tools for building software applications." APIs provide all of the building blocks that the programmer puts together.**

Task: Discuss database ODBC connectivity.

Database connectivity tools (ODBC, JDBC)

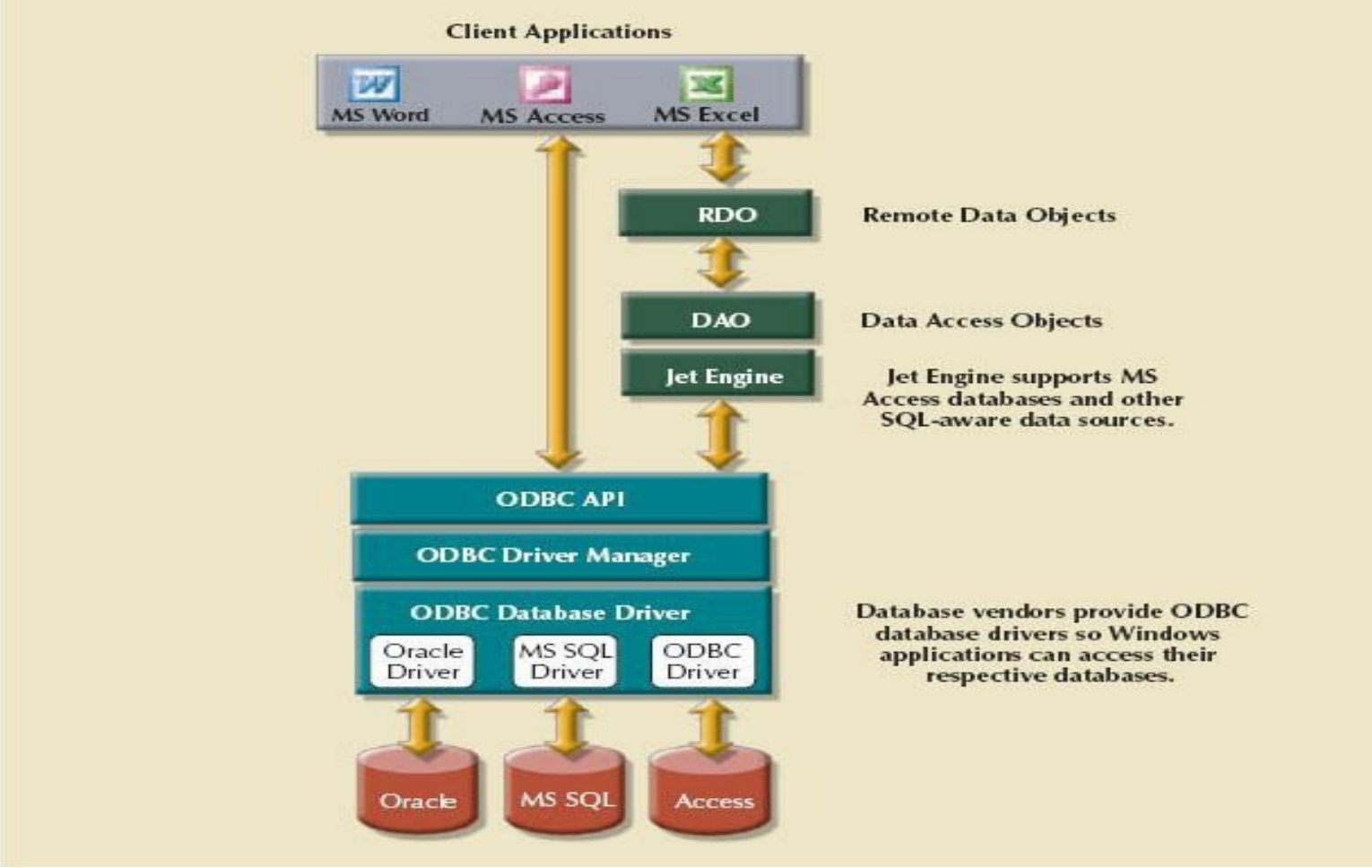
Open Database Connectivity(ODBC)

- Most operating environments, such as Windows, provide an API so that programmers can write applications consistent with the operating environment.
- APIs are good for users because they guarantee that all programs using a common API will have similar interfaces. That makes it easy for users to learn new programs.
- ODBC was the first widely adopted database middleware standard, and it enjoyed rapid adoption in Windows applications.
- As programming languages evolved, ODBC did not provide significant functionality beyond the ability to execute SQL to manipulate relational-style data.

Task: Discuss database ODBC connectivity.

Database connectivity tools (ODBC, JDBC)

Open Database Connectivity(ODBC)



Task: **Discuss database ODBC connectivity.**

Database connectivity tools (ODBC, JDBC)

Data Access Objects (DAO)

- Is an object-oriented API used to access MS Access, MS FoxPro, and dBase databases (using the Jet data engine) from Visual Basic programs.
- DAO provides an optimized interface that exposes programmers to the functionality of the Jet data engine, on which the MS Access database is based.
- The DAO interface can also be used to access other relational-style data sources.

Remote Data Objects (RDO)

- Is a higher-level, object-oriented application interface used to access remote database servers.
- RDO uses the lower-level DAO and ODBC for direct access to databases.
- RDO is optimized to deal with server-based databases such as MS SQL Server, Oracle, and DB2.

Task: Discuss database connectivity using DAO and RDO.

Database connectivity tools (ODBC, JDBC)

Open database connectivity(OBDC)- COMPONENTS

- **3 Main componets:**
  - **A high-level ODBC API for application programs to access ODBC functionality**
  - **A driver manager that is in charge of managing all database connections**
  - **An ODBC driver that communicates directly to the DBMS**
- **Defining a data source- is the first step in using ODBC. To define a data source, you must create a data source name (DSN) for it. To create a DSN, you need to provide the following:**
  - **An ODBC driver. identify the driver to use to connect to the data source; ODBC driver is normally provided by the database vendor; Microsoft provides several drivers that connect to most common databases. For example, if you are using an Oracle DBMS, you would select the Oracle ODBC driver provided by Oracle. Or, you could instead select the Microsoft-provided ODBC driver for Oracle.**

Task: Discuss database connectivity with ODBC.

Database connectivity tools (ODBC, JDBC)

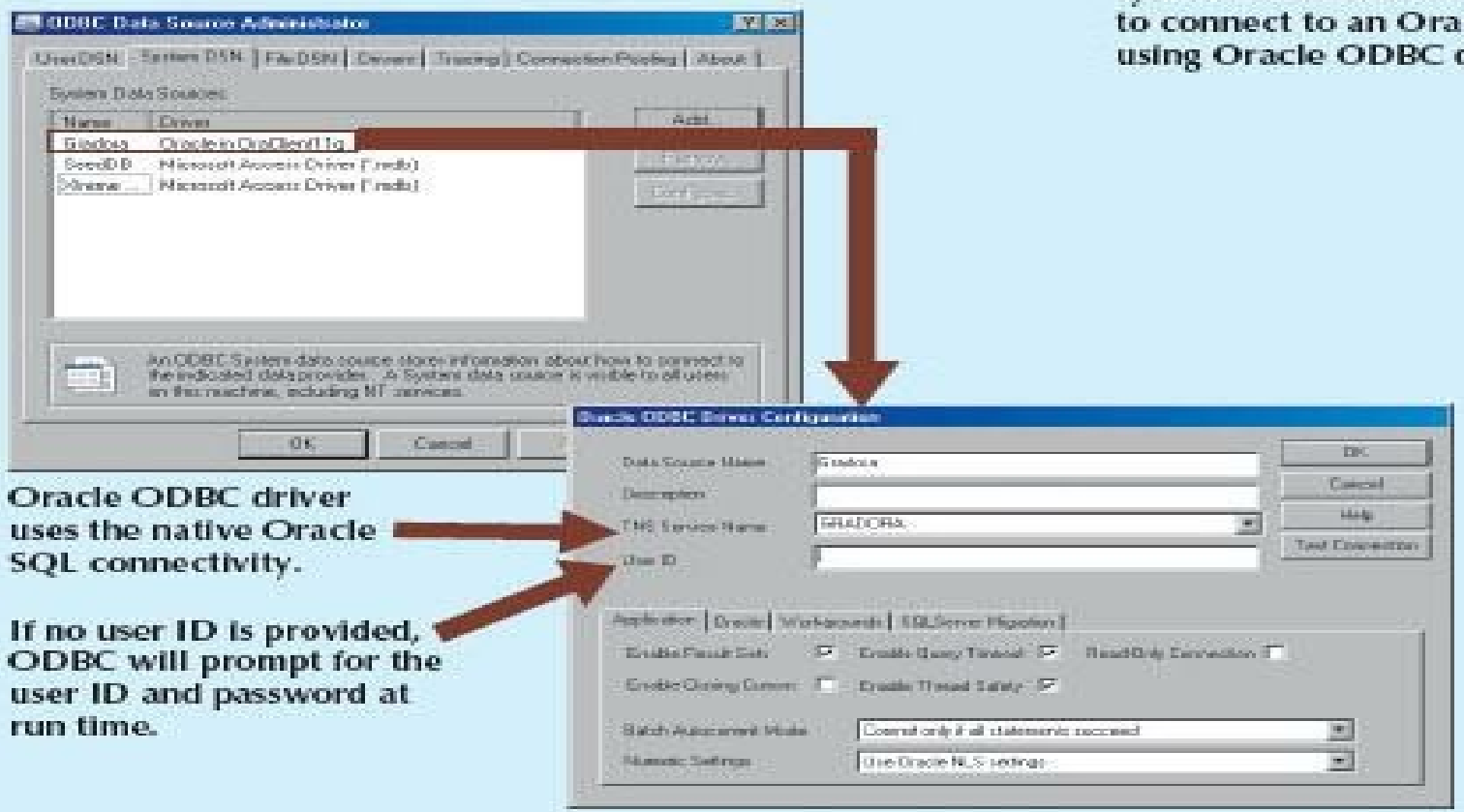
Open database connectivity(OBDC)- COMPONENTS

- **Defining a data source-** is the first step in using ODBC. To define a data source, you must create a data source name (DSN) for it. To create a DSN, you need to provide the following:
  - **A name**. A unique name by which the data source will be known to ODBC, and therefore to applications. ODBC offers two types of data sources: user and system. User data sources are available only to the user. System data sources are available to all users, including operating system services.
  - **ODBC driver parameters.** Most ODBC drivers require specific parameters to establish a connection to the database. For example, if you are using an MS Access database, you must point to the location of the MS Access file and then provide a username and password if necessary.
  - If you are using a DBMS server, you must provide the server name, the database name, the username, and the password needed to connect to the database.

Task: Discuss database connectivity using ODBC.

Database connectivity tools (ODBC, JDBC)

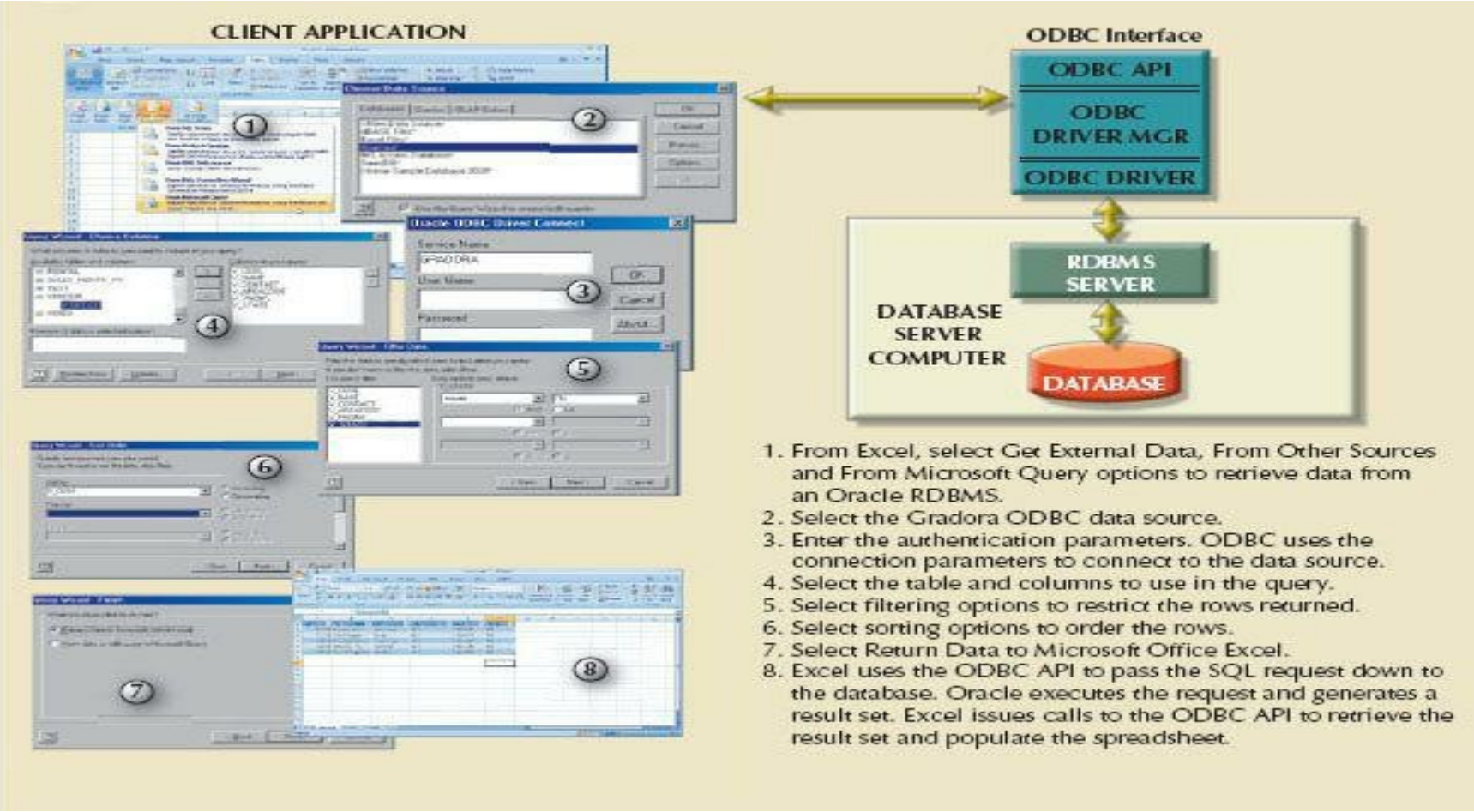
Open database connectivity(OBDC)- COMPONENTS



Task: **Discuss database connectivity using ODBC.**

Database connectivity tools (ODBC, JDBC)

Open database connectivity(OBDC)- CONNECT TO ACCESS



Task: **Discuss database connectivity using ODBC: EXCEL TO ORACLE.**

Database connectivity tools (ODBC, JDBC)

JAVA DATABASE CONNECTIVITY (JDBC)

- Java is an object-oriented programming language now under Oracle, that runs on top of Web browser software.
- Java is one of the most common programming languages for Web development.
- Java is a “write once, run anywhere” language.
- Write a Java application once and then run it in multiple environments without any modification.
- Java run-time environments exist for most operating systems, from computers to handheld mobile devices to TV set-top boxes. Another advantage of using Java is its “on-demand” architecture. When a Java application loads, it can dynamically download all its modules or required components via the Internet.

Task: Discuss database connectivity using JDBC.

Database connectivity tools (ODBC, JDBC)

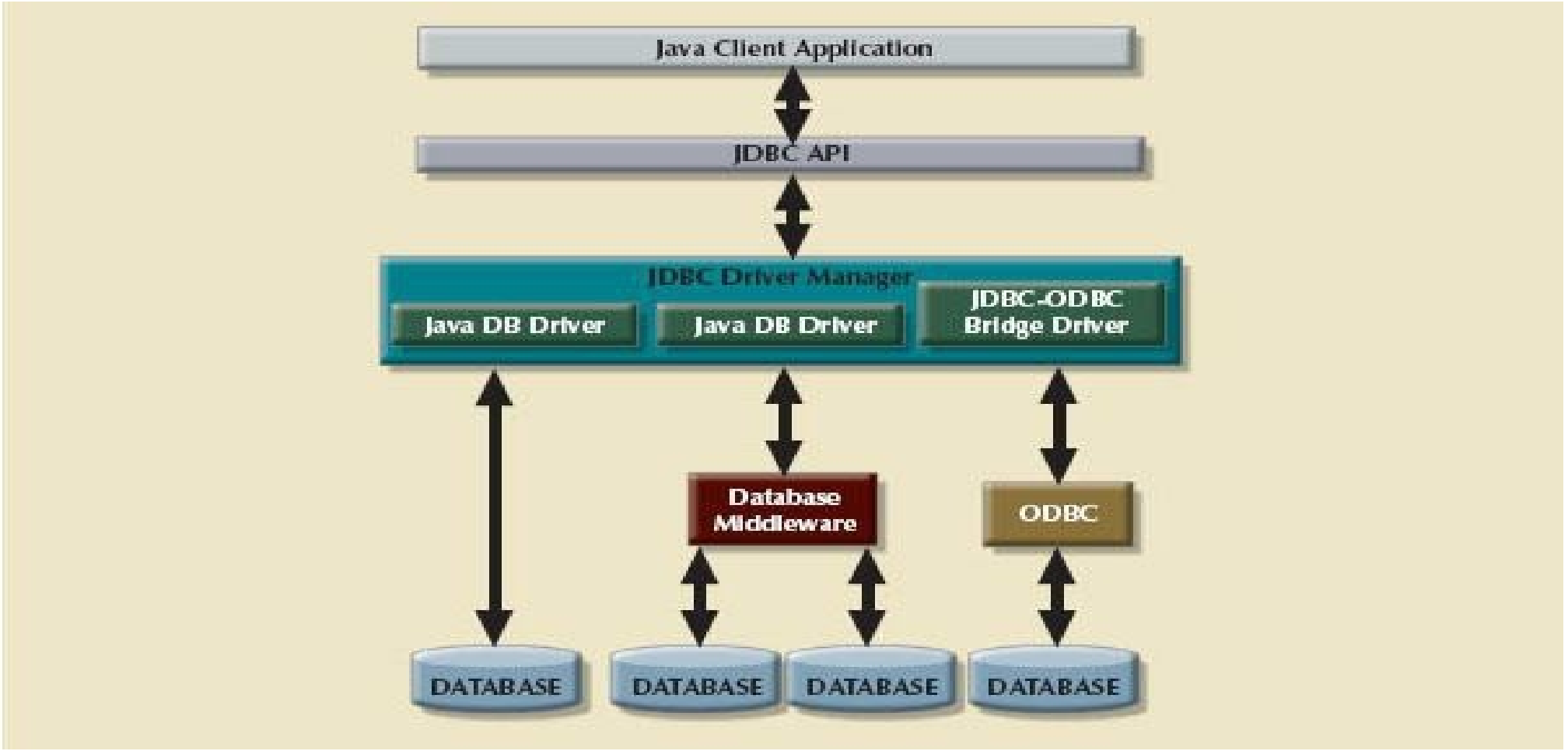
JAVA DATABASE CONNECTIVITY (JDBC)

- **Java Database Connectivity (JDBC) is an application programming interface that allows a Java program to interact with a wide range of data sources, including relational databases, tabular data sources, spreadsheets, and text files.**
- **JDBC allows a Java program to establish a connection with a data source, prepare and send the SQL code to the database server, and process the result set.**
- **One main advantage of JDBC is that it allows a company to leverage its existing investment in technology and personnel training.**
- **JDBC allows programmers to use their SQL skills to manipulate the data in the company's databases.**
- **As a matter of fact, JDBC allows direct access to a database server or access via database middleware.**
- **JDBC also provides a way to connect to databases through an ODBC driver.**

Task: **Discuss database connectivity using JDBC.**

Database connectivity tools (ODBC, JDBC)

JAVA DATABASE CONNECTIVITY (JDBC)



Task: **Discuss JDBC architecture.**

Database connectivity tools (ODBC, JDBC)

JAVA DATABASE CONNECTIVITY (JDBC)

- **Advantage of JDBC over other middleware is that it requires no configuration on the client side.**
- **The JDBC driver is automatically downloaded and installed as part of the Java applet download.**
- **Since Java is a Web-based technology, applications can connect to a database directly using a simple URL.**
- **Once the URL is invoked, the Java architecture comes into play, the necessary applets are downloaded to the client (including the JDBC database driver and all configuration information), and then the applets are executed securely in the client's run-time environment.**

Task: **Discuss database connectivity using JDBC.**

Database connectivity tools (ODBC, JDBC)

DATABASE INTERNET CONNECTIVITY

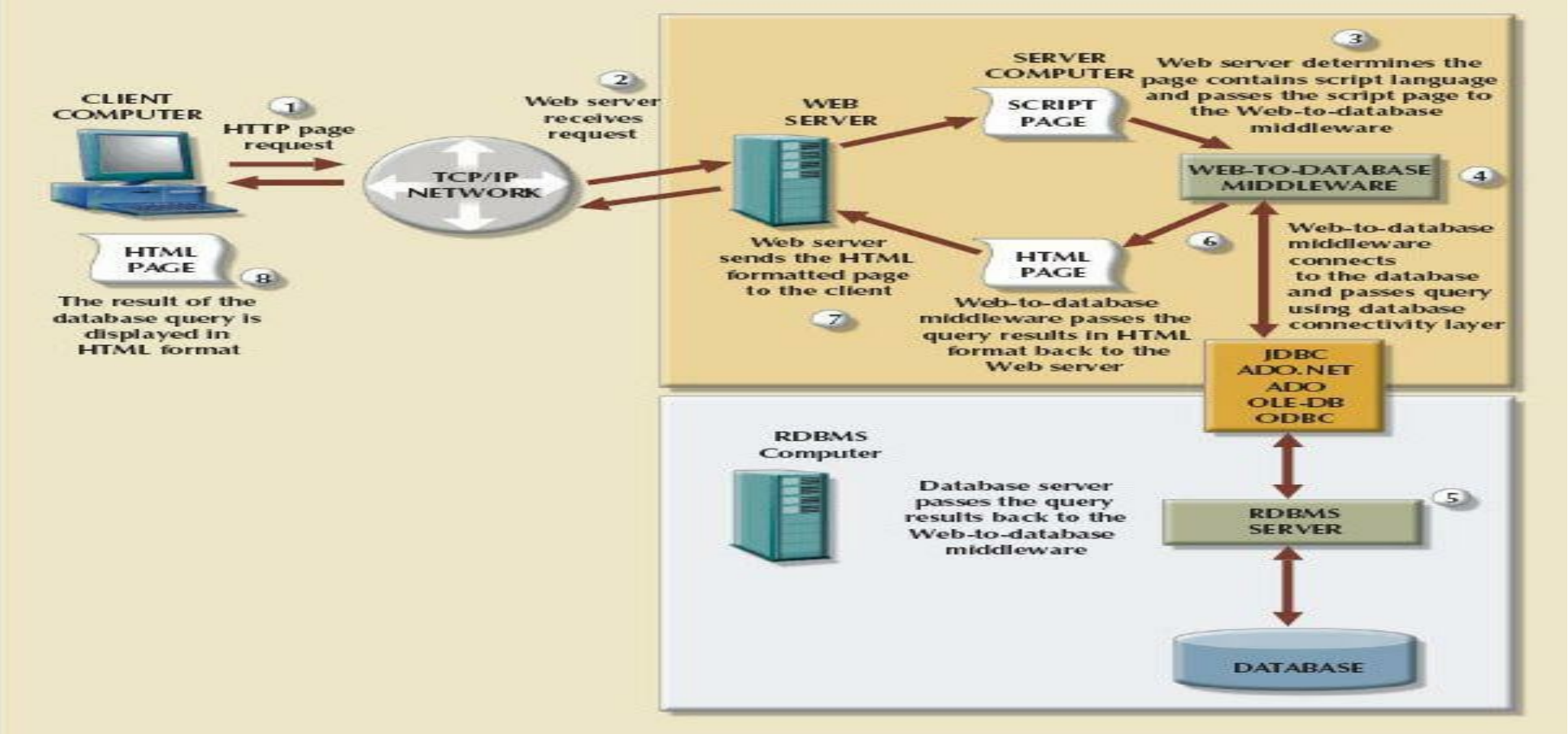
- **Millions of people all over the world access the Internet and connect to databases via Web browsers or data services. For example, they can use a smart phone applet to get weather information.**

A server-side extension is a program that interacts directly with the Web server to handle specific types of requests.

- **The server-side extension program retrieves the data from databases and passes the retrieved data to the Web server, which in turn sends the data to the client's browser for display.**
- **The server-side extension makes it possible to retrieve and present the query results, but more importantly, it provides its services to the Web server in a way that is totally transparent to the client browser.**
- **Server-side extension adds significant functionality to the Web server, and to the Internet.**

Task: Discuss database internet connectivity.

Database connectivity tools (ODBC, JDBC): DATABASE INTERNET CONNECTIVITY



Task: **Discuss database internet connectivity.**

Database connectivity tools (ODBC, JDBC)- DATABASE INTERNET CONNECTIVITY

Trace the Web-to-database middleware actions in slide above:

- **The client browser sends a page request to the Web server.**
- **The Web server receives and passes the request to the Web-to-database middleware for processing. Generally, the requested page contains some type of scripting language to enable the database interaction.**
- **The Web-to-database middleware reads, validates, and executes the script. In this case, it connects to the database and passes the query using the database connectivity layer.**
- **The database server executes the query and passes the result back to the Web-to-database middleware.**
- **The Web-to-database middleware compiles the result set, dynamically generates an HTML-formatted page that includes the data retrieved from the database, and sends it to the Web server.•The Web server returns the just-created HTML page, which now includes the query result, to the client browser.**
- **The client browser displays the page on the local computer.**

Task: Discuss database internet connectivity.

Week 11 exercises

1) Describe the Native SQL connectivity.

2) Describe the Microsoft's Open Database Connectivity (ODBC).

3) Describe the Java Database Connectivity (JDBC)

4) Describe the Database Internet Connectivity.

Week 11 Session References

- **[Course Text] Carlos Coronel, Steven Morris, Peter Rob and Keeley Crockett Database Principles: Fundamentals of Design, Implementation, and Management, 14th Edition, 2022, ISBN-13978-0357673034.**

- **Thomas M. Connolly, Carolyn E. Begg (2021). Database Systems: A Practical Approach to Design, Implementation, and Management. Published by Pearson (July 14th 2021). ISBN-13: 9780137517053**

Thank You