Database Applications and Web-Enabled Databases

University of California, Berkeley

School of Information

IS 257: Database Management

Announcements

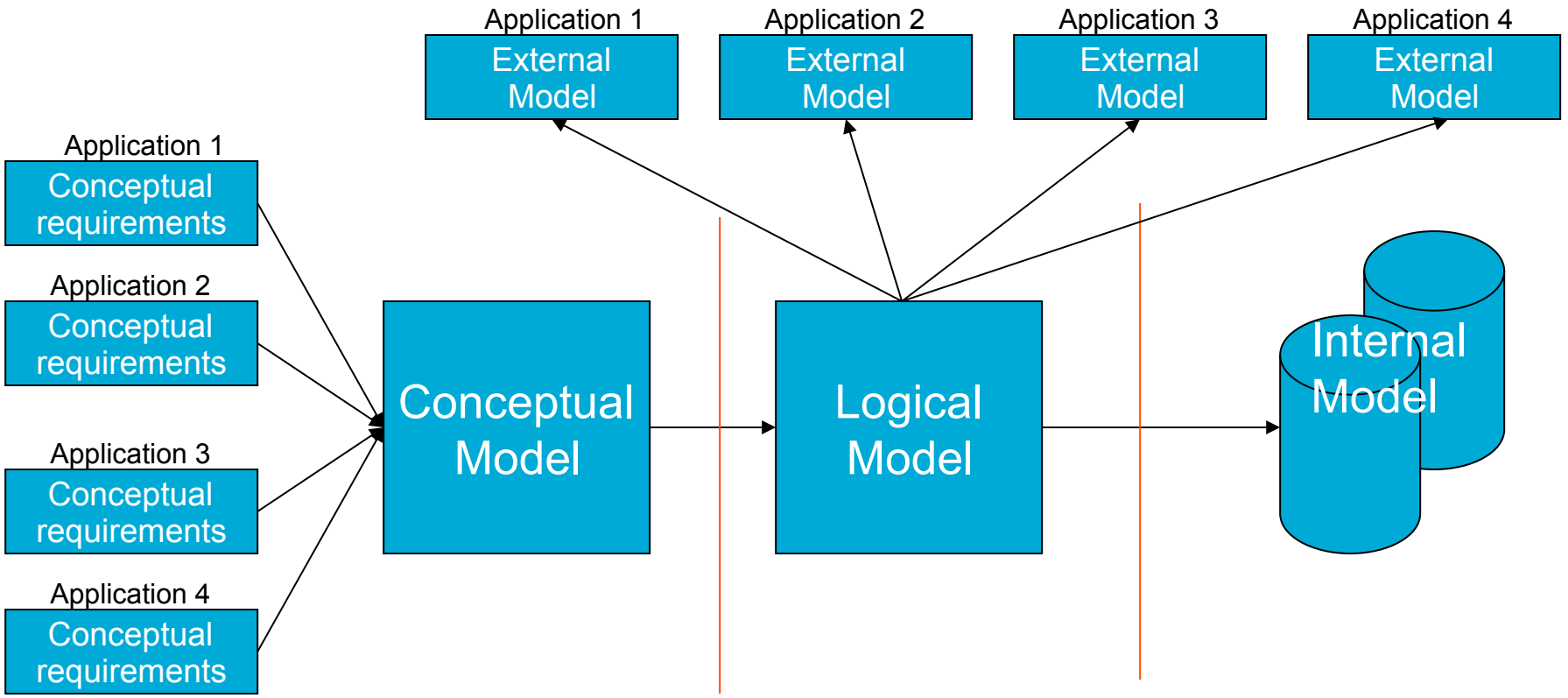
Lecture Outline

- Review
  - Database design review
  - Introduction to SQL and MySQL
- Application Development in Access
- Databases for Web Applications – Overview

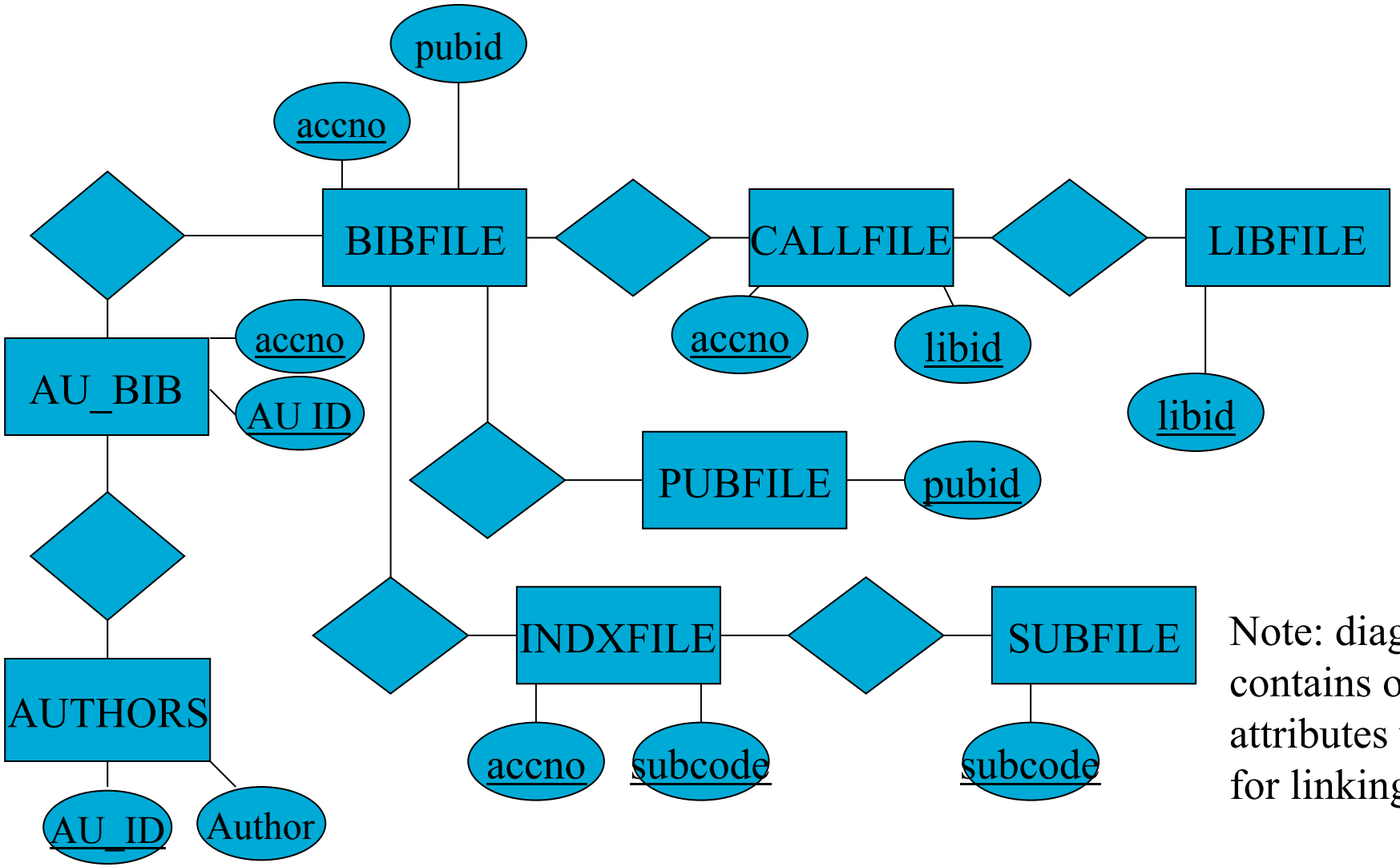
Lecture Outline

- Review
  - Database design review
  - Introduction to SQL & MySQL
- Application Development in Access
- Databases for Web Applications – Overview

Database Design Process



Cookie ER Diagram



Note: diagram contains only attributes used for linking

Logical Model: Mapping to Relations

- Take each entity
  - Authors
  - BIBFILE
  - LIBFILE
  - CALLFILE
  - SUBFILE
  - PUBFILE
  - INDXFILE
  - AU_BIB
- And make it a table...

Physical Model: SQL for Creation

- We looked at how an SQL "script" could be created that would create each of the relational tables, define primary keys and indexes and load data into the database

MySQL Data Types

- MySQL supports all of the standard SQL numeric data types. These types include the exact numeric data types (INTEGER, SMALLINT, DECIMAL, and NUMERIC), as well as the approximate numeric data types (FLOAT, REAL, and DOUBLE PRECISION). The keyword INT is a synonym for INTEGER, and the keyword DEC is a synonym for DECIMAL
- Numeric (can also be declared as UNSIGNED)
  - BIT(n) (variable field of n bits)
  - BOOL or BOOLEAN (internally is TINYINT with value of 0 for FALSE)
  - TINYINT (1 byte)
  - SMALLINT (2 bytes)
  - MEDIUMINT (3 bytes)
  - INTEGER (4 bytes)
  - INT (4 bytes - Synonym)
  - BIGINT (8 bytes)
  - NUMERIC or DECIMAL (Packed - up to 65 digits - DEC, FIXED synonyms)
  - FLOAT
  - DOUBLE (or DOUBLE PRECISION)
  - **SERIAL** = BIGINT UNSIGNED NOT NULL AUTO_INCREMENT UNIQUE

MySQL Data Types

- The date and time types for representing temporal values are DATETIME, DATE, TIMESTAMP, TIME, and YEAR. Each temporal type has a range of legal values, as well as a "zero" value that is used when you specify an illegal value that MySQL cannot represent
  - DATETIME '0000-00-00 00:00:00'
  - DATE '0000-00-00'
  - TIMESTAMP (4.1 and up) '0000-00-00 00:00:00'
  - TIMESTAMP (before 4.1) 00000000000000
  - TIME '00:00:00'
  - YEAR 0000

MySQL Data Types

- The string types are CHAR, VARCHAR, BINARY, VARBINARY, BLOB, TEXT, ENUM, and SET
- Maximum length for CHAR is 255 and VARCHAR is **65,535** (limited by row size)

Value	CHAR(4)	Storage	VARCHAR(4)	Storage
""	" "	4	""	1
"ab"	"ab "	4	"ab"	3
"abcd"	"abcd"	4	"abcd"	5
"abcdefg"	"abcd"	4	"abcd"	5

- **For longer things there is BLOB and TEXT**

MySQL Data Types

- A **BLOB** is a binary large object that can hold a variable amount of data.
- The four BLOB types are TINYBLOB, BLOB, MEDIUMBLOB, and LONGBLOB. These differ only in the maximum length of the values they can hold
- The four TEXT types are TINYTEXT, TEXT, MEDIUMTEXT, and LONGTEXT. These correspond to the four BLOB types and have the same maximum lengths and storage requirements
- TINY=1byte, BLOB and TEXT=2bytes, MEDIUM=3bytes, LONG=4bytes

MySQL Data Types

- BINARY and VARBINARY are like CHAR and VARCHAR but are intended for binary data of 255 bytes or less
- ENUM is a list of values that are stored as their addresses in the list
  - For example, a column specified as ENUM('one', 'two', 'three') can have any of the values shown here. The index of each value is also shown:
    - **Value = Index**
    - NULL = NULL
    - ‘’ = 0
    - 'one’ = 1
    - ‘two’ = 2
    - ‘three’ = 3
  - An enumeration can have a maximum of 65,535 elements.

MySQL Data Types

- The final string type (for this version) is a SET
- A SET is a string object that can have zero or more values, each of which must be chosen from a list of allowed values specified when the table is created.
- SET column values that consist of multiple set members are specified with members separated by commas (‘,’)
- For example, a column specified as SET('one', 'two') NOT NULL can have any of these values:
  - ''
  - 'one'
  - 'two'
  - 'one,two ‘
- A set can have up to 64 member values and is stored as an 8byte number

ALTER Table

- ALTER TABLE table-name ADD COLUMN col_name col_definition;
- … DROP COLUMN col_name;
- … CHANGE col_name new_col_definition;
- Adds/removes a new column from an existing database table
- Many other options for adding constraints (like NOT NULL, or PRIMARY KEY), etc.

INSERT

- INSERT INTO table-name (attr1, attr4, attr5,…, attrK) VALUES (“val1”, val4, val5,…, “valK”);
- Adds a new row(s) to a table.
- INSERT INTO table-name (attr1, attr4, attr5,…, attrK) VALUES SELECT ...

Creating a new table data from existing tables

- Syntax:
  - INSERT INTO tablename (attr1, attr2, attr3)

    SELECT [DISTINCT] xattr1, xattr2, xattr3 FROM rel1 r1, rel2 r2,… rel3 r3 WHERE condition1 {AND | OR} condition2 ORDER BY attr1 [DESC], attr3 [DESC]

tablename has to previously exist for this to work in MySQL…

DELETE

- DELETE FROM table-name WHERE <where clause>;
- Removes rows from a table.

UPDATE

- UPDATE tablename SET attr1=newval, attr2 = newval2 WHERE <where clause>;
- changes values in existing rows in a table (those that match the WHERE clause).

DROP Table

- DROP TABLE tablename;
- Removes a table from the database.

CREATE INDEX

- CREATE [UNIQUE|FULLTEXT|SPATIAL] INDEX indexname indextype ON tablename (attr1 [ASC|DESC][, attr2 [ASC|DESC], ...]) [USING [BTREE|HASH|RTREE]]

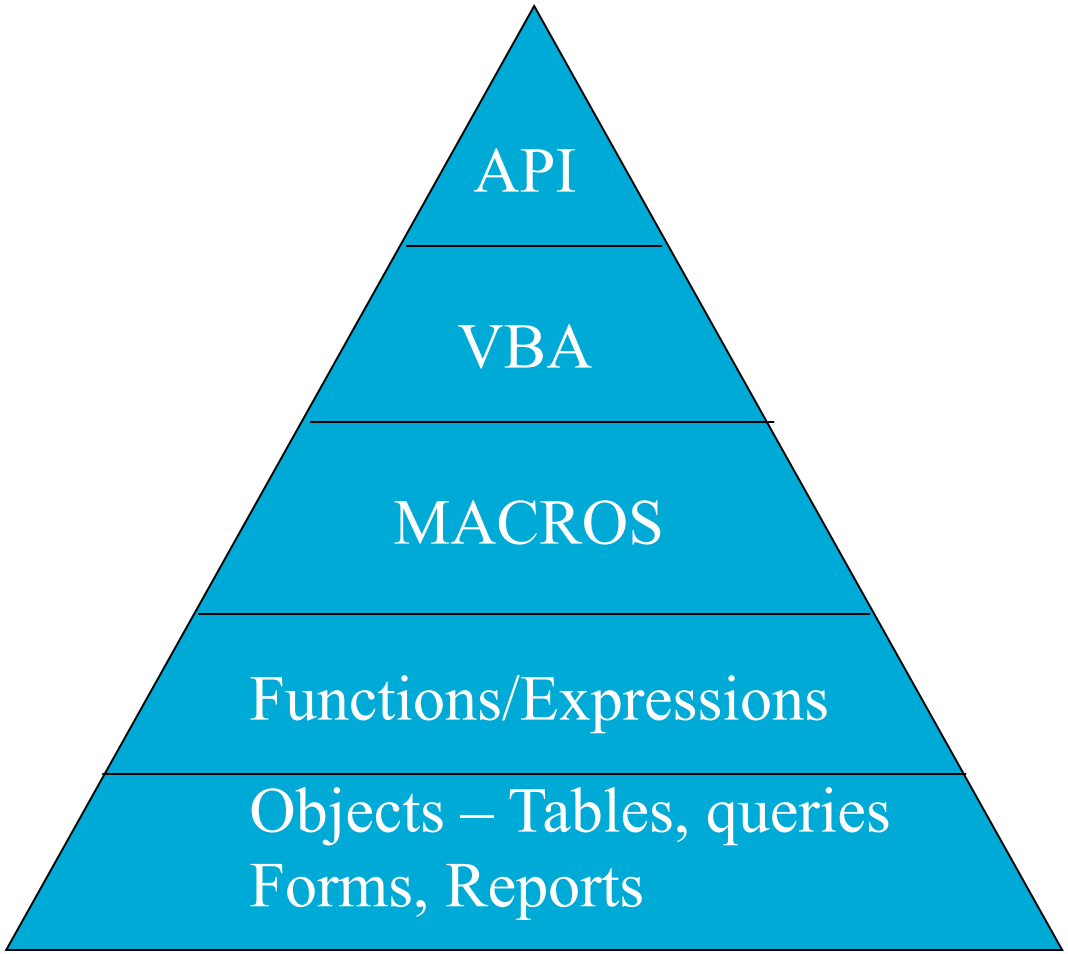
Lecture Outline

- Review
  - Introduction to SQL and MySQL
- Application Development in Access
- Databases for Web Applications – Overview

Database Applications

- Generally, end-users of database data probably do not want to learn SQL in order to access the information in the database
- Instead, they would prefer to use a familiar PC or Web interface that uses the graphical conventions and behaviors that they are familiar with
- Today we will look briefly at PC –style client applications using systems like Access and Web-based systems

Access Usability Hierarchy



From McFadden
Chap. 10

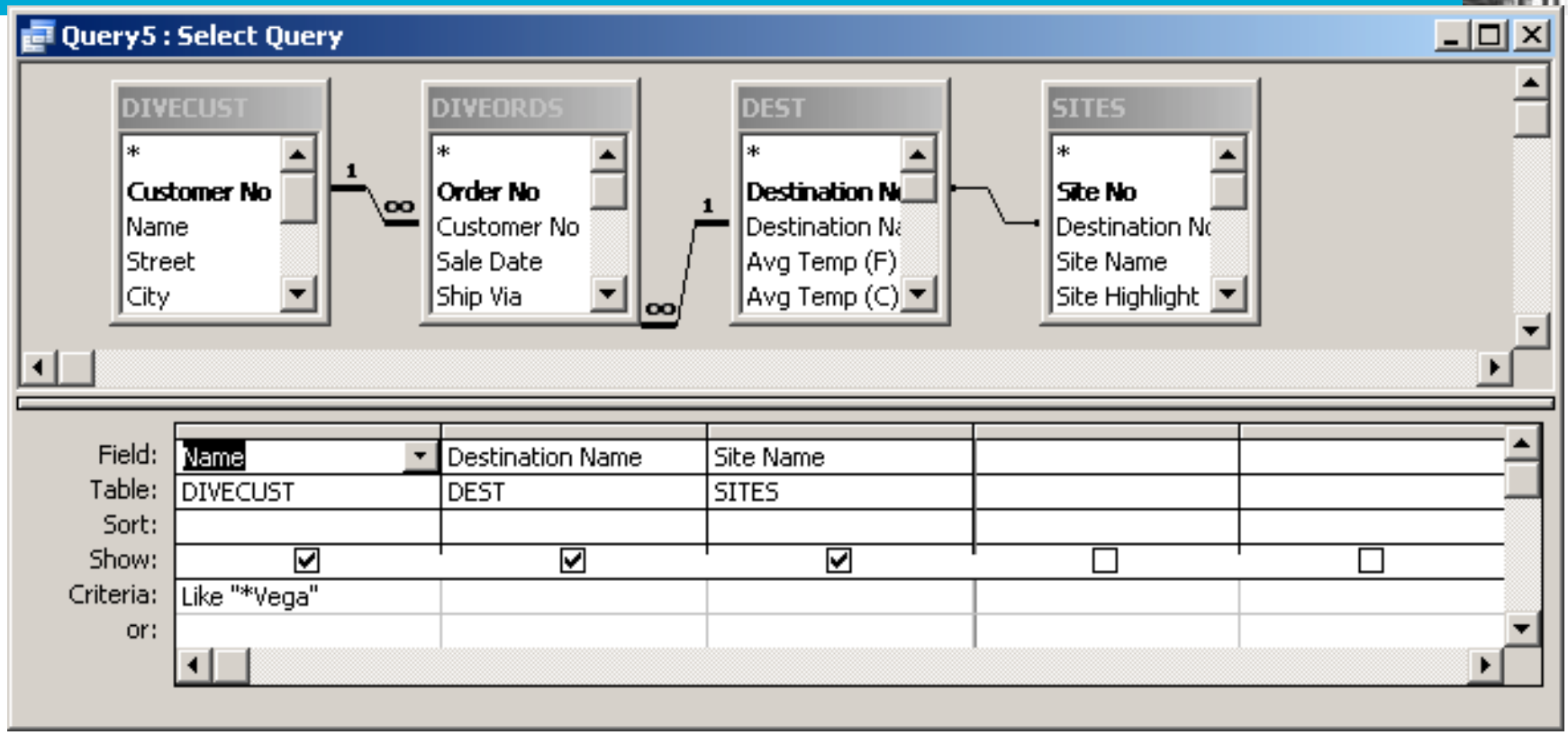
Examples

- Access OBJECT level
  - QBE querying
- Building Application interfaces
  - User wants "point and click" and forms to fill in, not a Query editing screen or wizard
  - How to build them
    - Drag and drop as in Access
    - Programming Languages
    - 4th Generation languages (more on these later)

Query-by-Example

- QBE was developed in the 1970s as a simpler to use interface for IBM mainframe databases
- In QBE the user puts parts of what they want to get from the database into a form similar to what the output will look like
- The Query Design View in Access is an example of QBE

Access Query Interface…



What sites might Lorraine Vega dive on her trip? – SQL generated…

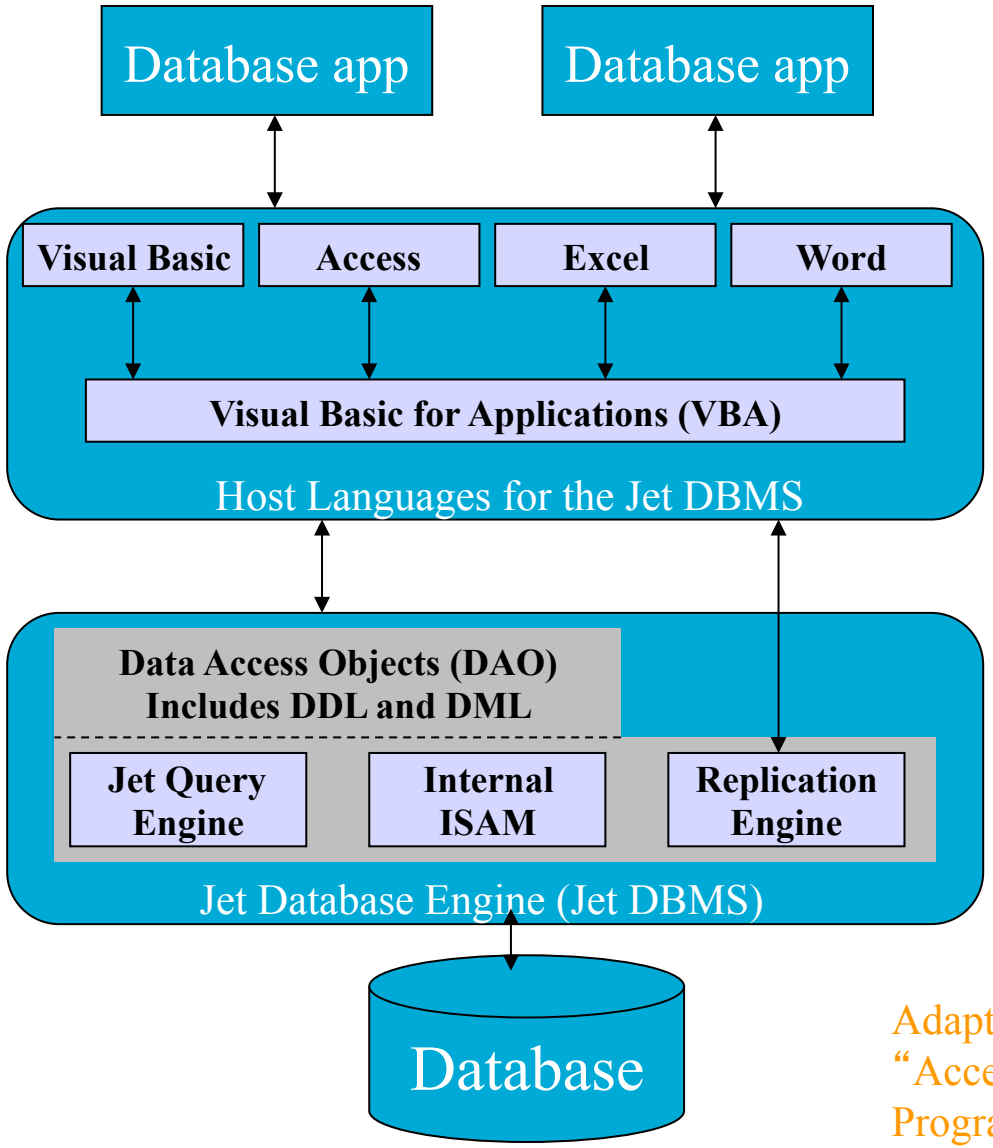
```
SELECT DIVECUST.Name, DEST.[Destination Name], SITES.[Site Name]
FROM ((DIVECUST INNER JOIN DIVEORDS
ON DIVECUST.[Customer No] = DIVEORDS.[Customer No])
INNER JOIN DEST ON DIVEORDS.Destination = DEST.[Destination Name])
INNER JOIN SITES ON DEST.[Destination No] = SITES.[Destination No]
WHERE (((DIVECUST.Name) Like "*Vega"));
```

Access Query Interface

- Output is generated in a window…

Query5		
Name	**Destination Name**	**Site Name**
Lorraine Vega	Cozumel	Palancar Reef
Lorraine Vega	Cozumel	Santa Rosa Reef
Lorraine Vega	Cozumel	Chancanab Reef
Lorraine Vega	Cozumel	Punta Sur
Lorraine Vega	Cozumel	Yocab Reef

The MS JET Database Engine

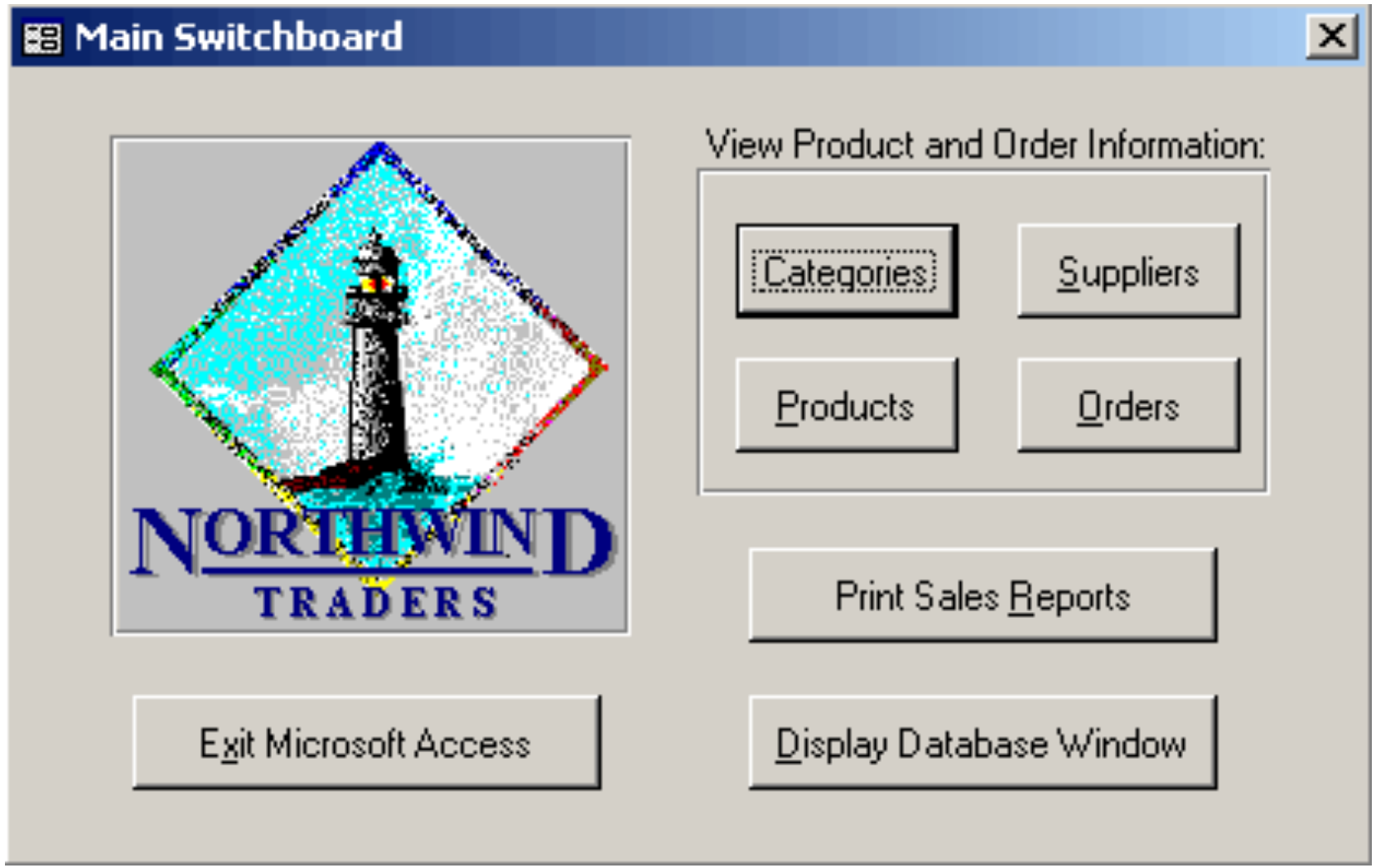


Adapted from Roman, "Access Database Design and Programming"

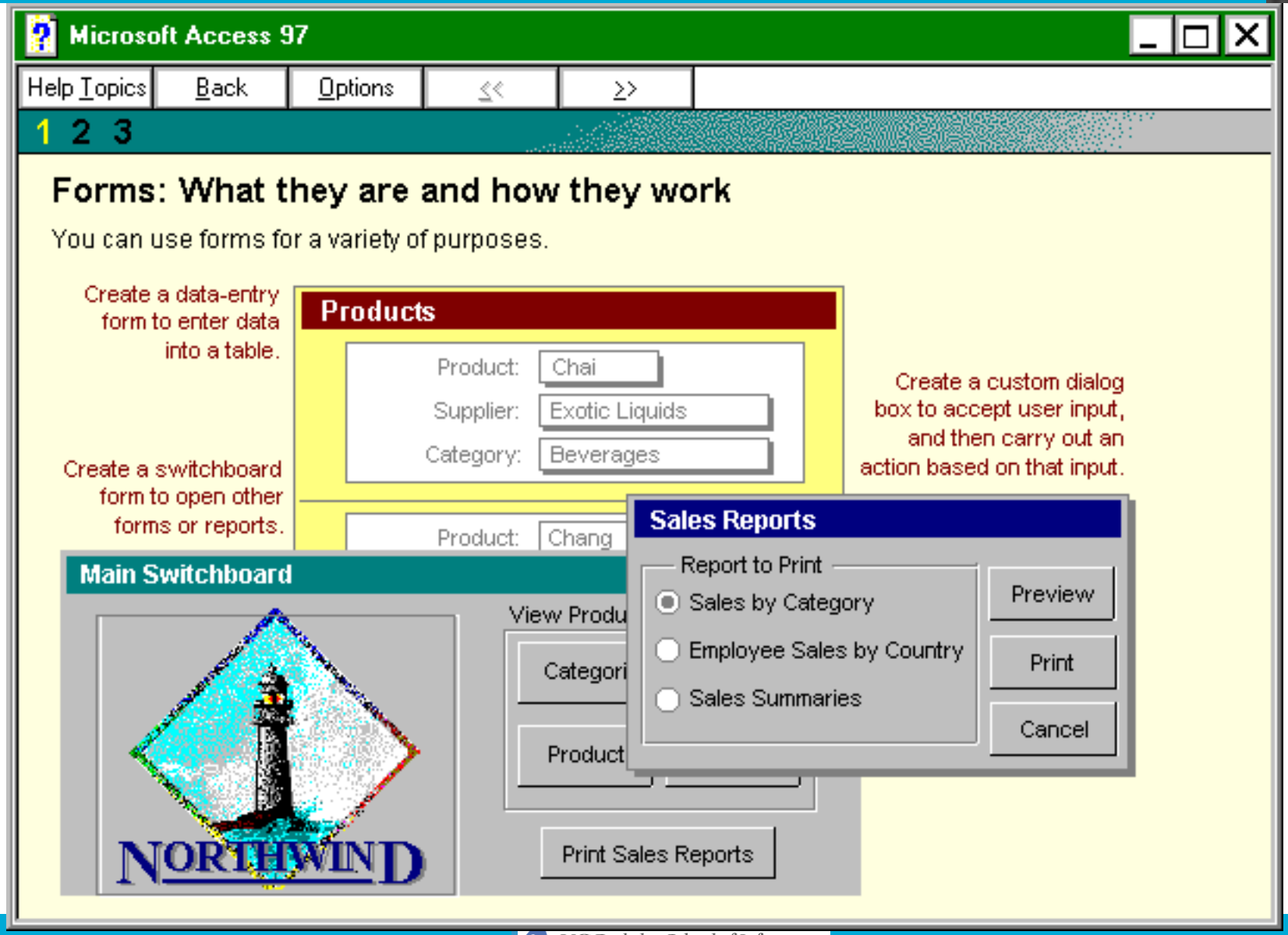
Using Access for Applications

- Forms
- Reports
- Macros
- VBA programming
- Application framework
- HTML Pages

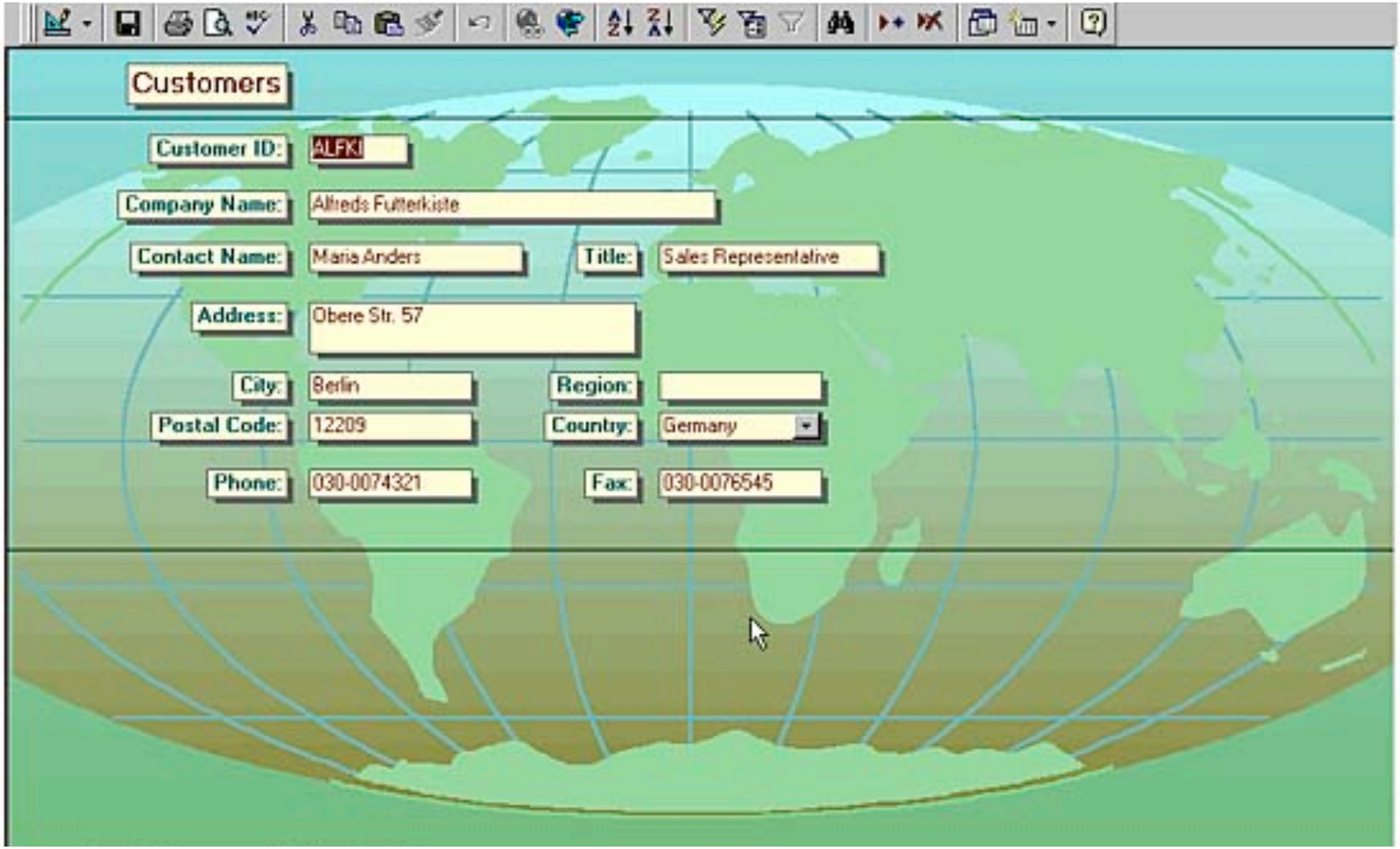
Access Applications



Access Forms



Access Forms



Forms – including query results

Categories

NORTHWIND TRADERS

Categories
Products
Suppliers
Orders
Employees
Customers
Customer Phone List
Customer Orders
Quaterly Orders
Reports

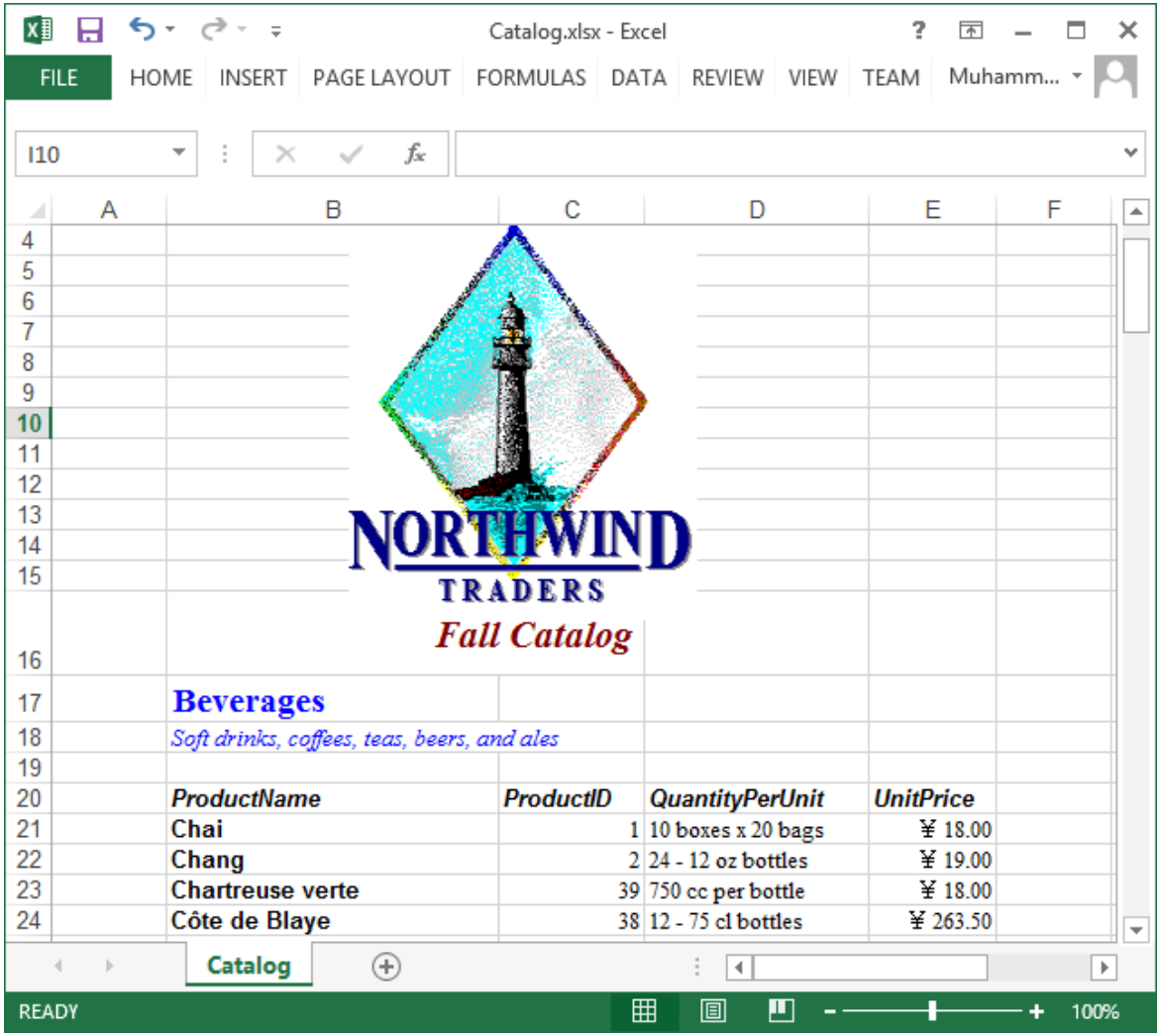
Name: Beverages

Description: Soft drinks, coffees, teas, beers, and ales

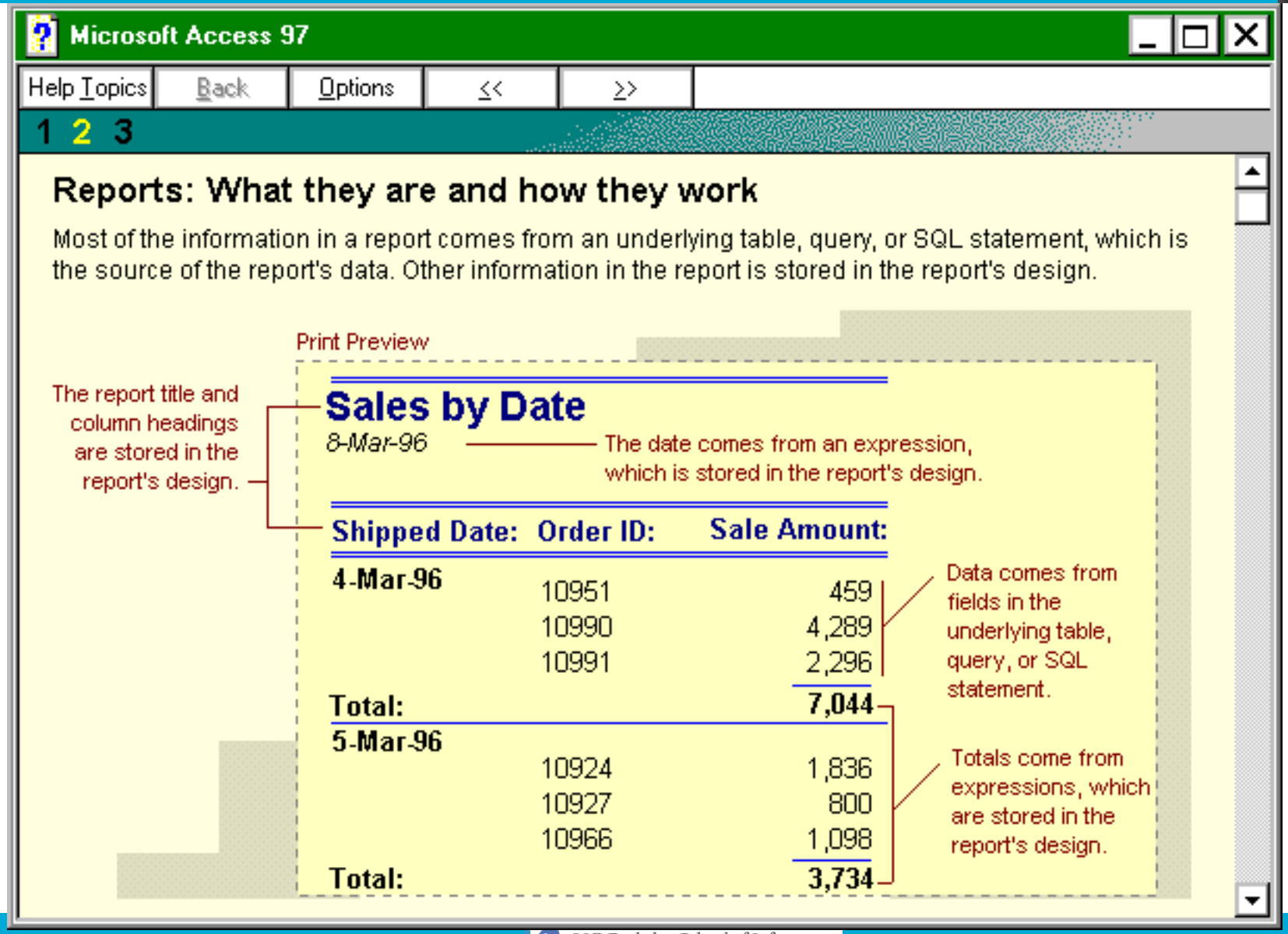
Picture:

Name	QuantityPerUnit	UnitPrice	Discontinued
Chai	10 boxes x 20 b...	18.0000	☐
Chang	24 - 12 oz bottles	19.0000	☐
Guaraná Fantástica	12 - 355 ml cans	4.5000	☑
Sasquatch Ale	24 - 12 oz bottles	14.0000	☐
Steeleye Stout	24 - 12 oz bottles	18.0000	☐
Côte de Blaye	12 - 75 cl bottles	263.5000	☐
Chartreuse verte	750 cc per bottle	18.0000	☐
Ipoh Coffee	16 - 500 g tins	46.0000	☐
Laughing Lumberjack Lager	24 - 12 oz bottles	14.0000	☐
Outback Lager	24 - 355 ml bott...	15.0000	☐
Rhönbräu Klosterbier	24 - 0.5 l bottles	7.7500	☐
Lakkalikööri	500 ml	18.0000	☐

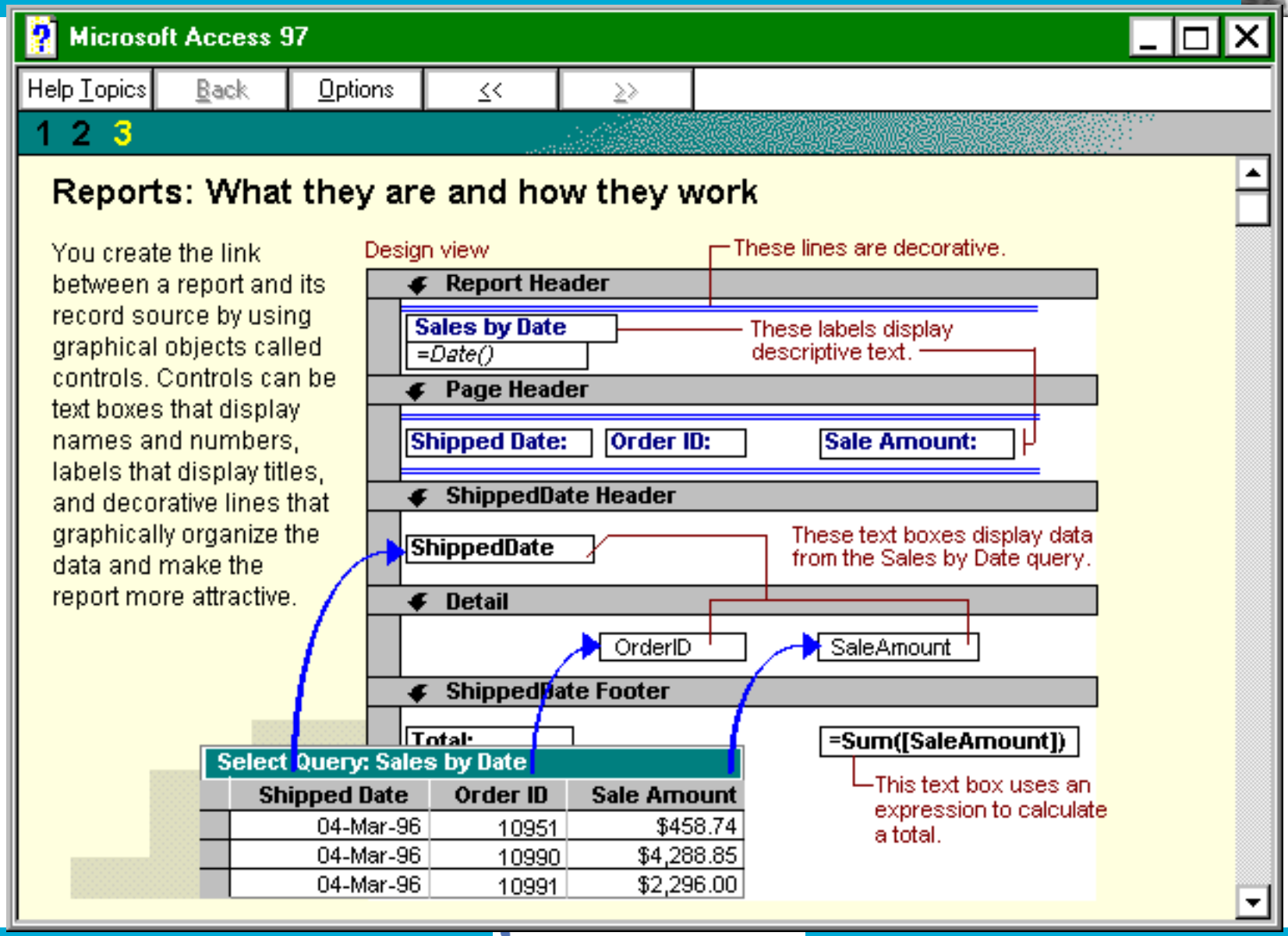
Form Layout and Design



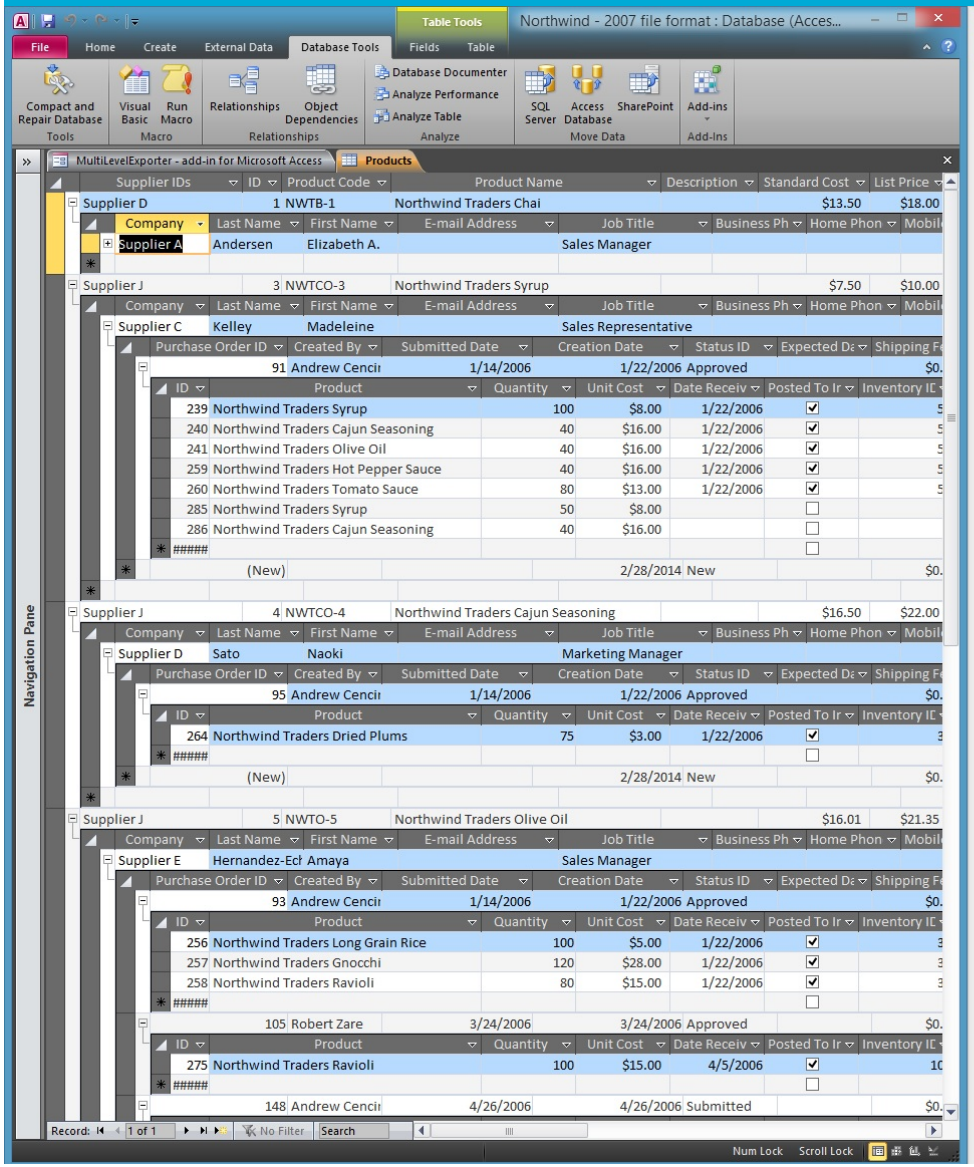
Reports

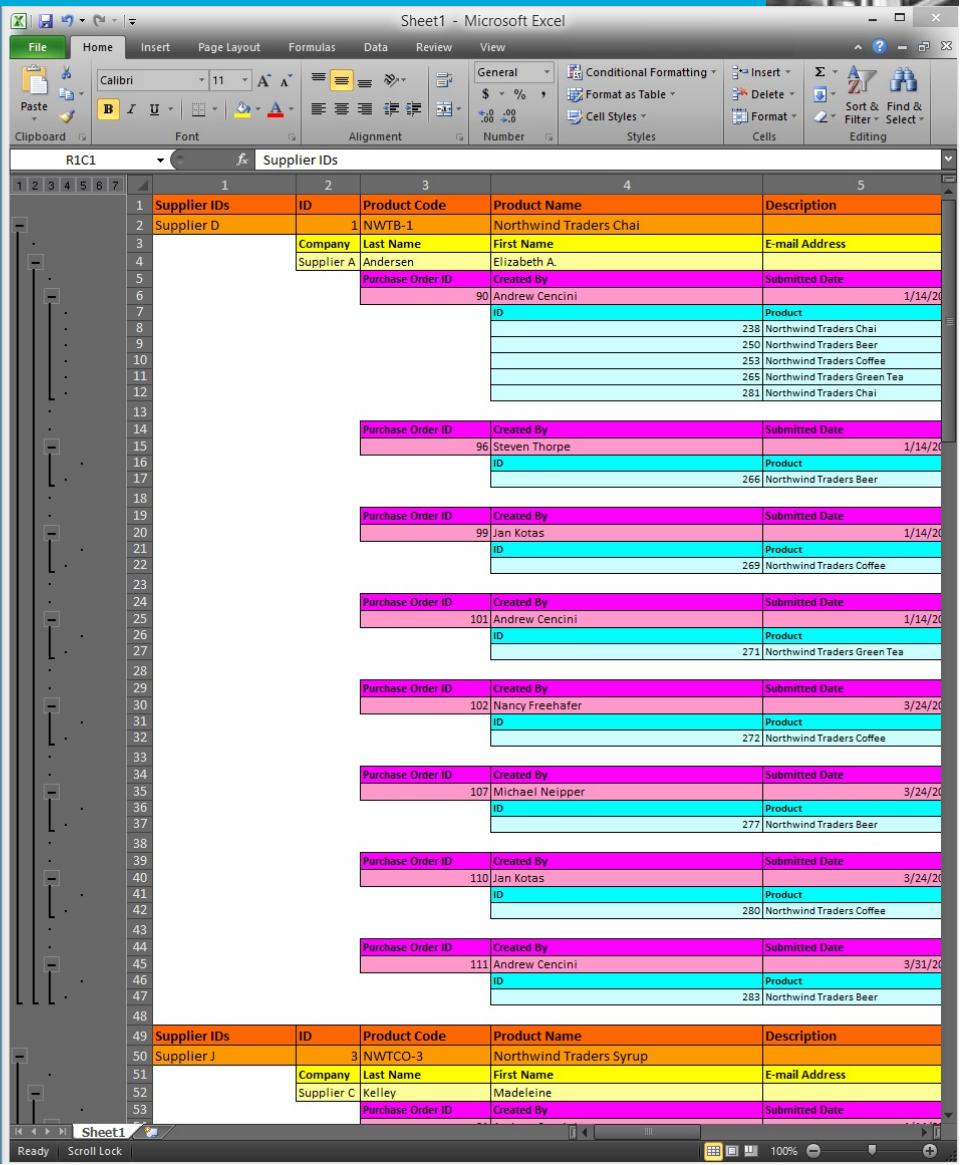

Microsoft Access 97
Help Topics
Back
Options
<<
>>
1 2 3
Reports: What they are and how they work
Most of the information in a report comes from an underlying table, query, or SQL statement, which is the source of the report's data. Other information in the report is stored in the report's design.
Print Preview
The report title and column headings are stored in the report's design.
Sales by Date
8-Mar-96
The date comes from an expression, which is stored in the report's design.
Shipped Date: Order ID: Sale Amount:
4-Mar-96 10951 459
10990 4,289
10991 2,296
Data comes from fields in the underlying table, query, or SQL statement.
Total: 7,044
5-Mar-96 10924 1,836
10927 800
10966 1,098
Totals come from expressions, which are stored in the report's design.
Total: 3,734

Report Design

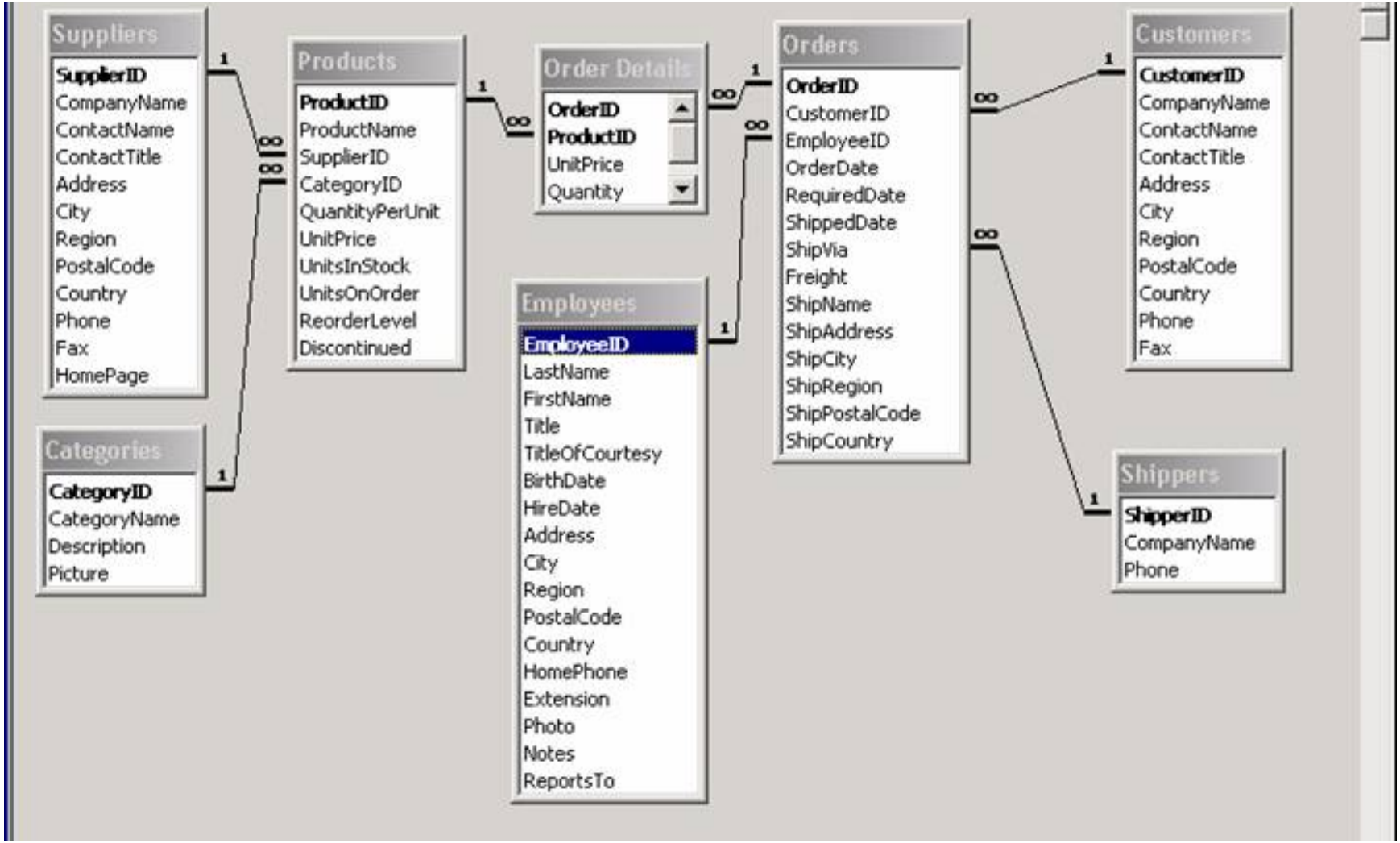


Reports – design and result





Access Relationships



Lecture Outline

- Review
  - Introduction to SQL
- Application Development in Access
- Databases for Web Applications – Overview

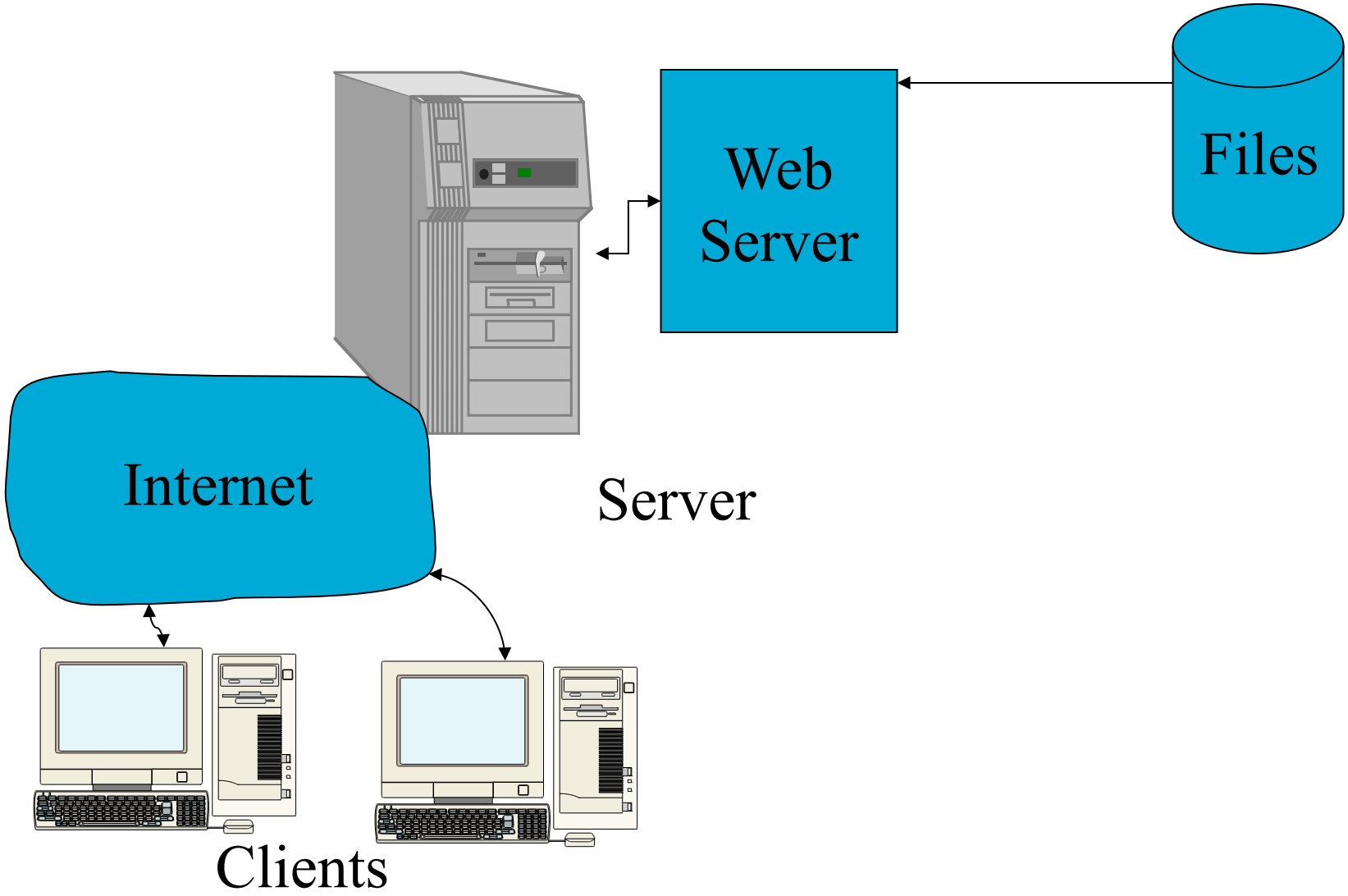
Overview

- Why use a database system for Web design and e-commerce?
- What systems are available?
- Pros and Cons of different web database systems?
- Text retrieval in database systems
- Search Engines for Intranet and Intrasite searching

Why Use a Database System?

- Simple Web sites with only a few pages don't need much more than static HTML files

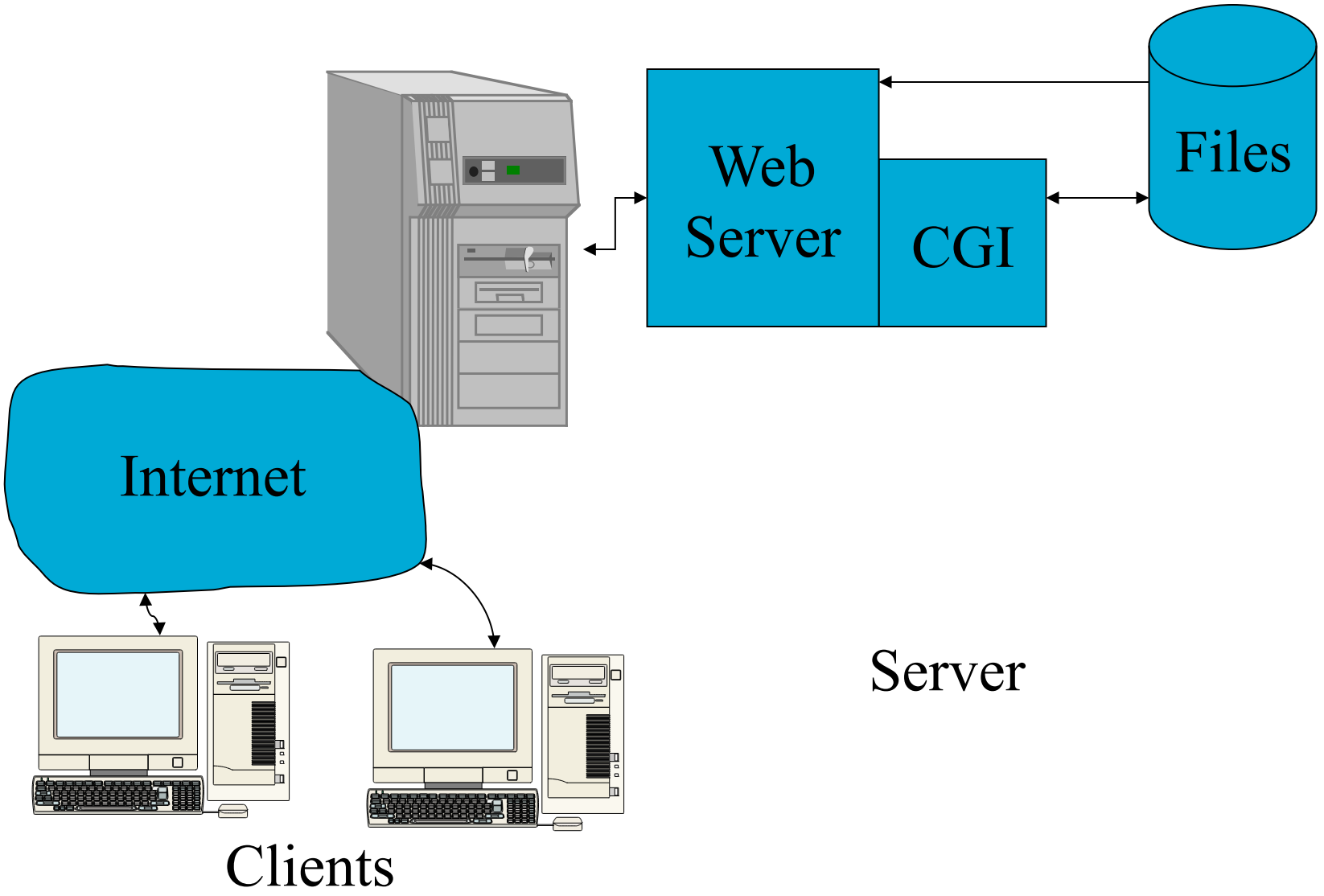
Simple Web Applications



Adding Dynamic Content to the Site

- Small sites can often use simple HTML and CGI scripts accessing data files to create dynamic content for small sites.

Dynamic Web Applications 1



Issues For Scaling Up Web Applications

- Performance
- Scalability
- Maintenance
- Data Integrity
- Transaction support

Performance Issues

- Problems arise as both the data to be managed and usage of the site grows.
  - Interpreted CGI scripts are inherently slower than compiled native programs
  - Starting CGI applications takes time for each connection
  - Load on the system compounds the problem
  - Tied to other scalability issues

Scalability Issues

- Well-designed database systems will permit the applications to scale to accommodate very large databases
  - A script that works fine scanning a small data file may become unusable when the file becomes large.
  - Issues of transaction workload on the site
    - Starting a separate copy of a CGI program for each user is NOT a scalable solution as the workload grows

Maintenance Issues

- Dealing with multiple data files (customer list, product list, customer orders, etc.) using CGI means:
  - If any data element in one of the files changes, all scripts that access that file must be rewritten
  - If files are linked, the programs must insure that data in all the files remains synchronized
  - A large part of maintenance will involve dealing with data integrity issues
  - Unanticipated requirements may require rewriting scripts

Data Integrity Constraint Issues

- These are constraints we wish to impose in order to protect the database from becoming inconsistent.
- Five basic types
  - Required data
  - attribute domain constraints
  - entity integrity
  - referential integrity
  - enterprise constraints

Transaction support

- Concurrency control (ensuring the validity of database updates in a shared multiuser environment).

No Concurrency Control: Lost updates

John

- Read account balance (balance = $1000)
- Withdraw $200 (balance = $800)
- Write account balance (balance = $800)

Marsha

- Read account balance (balance = $1000)
- Withdraw $300 (balance = $700)
- Write account balance (balance = $700)

ERROR!

Concurrency Control: Locking

- Locking levels
  - Database
  - Table
  - Block or page
  - Record
  - Field
- Types
  - Shared (S locks)
  - Exclusive (X locks)

Concurrency Control: Updates with X locking

John	Marsha
• Lock account balance • Read account balance (balance = $1000) • Withdraw $200 (balance = $800) • Write account balance (balance = $800) • Unlock account balance	• Read account balance (DENIED) • Lock account balance • Read account balance (balance = $800) • etc...

Concurrency Control: Deadlocks

John	Marsha
• Place S lock	
• Read account balance (balance = $1000)	• Place S lock
	• Read account balance (balance = $1000)
• Request X lock (denied)	
• wait ...	• Request X lock (denied)
	• wait...

Deadlock!

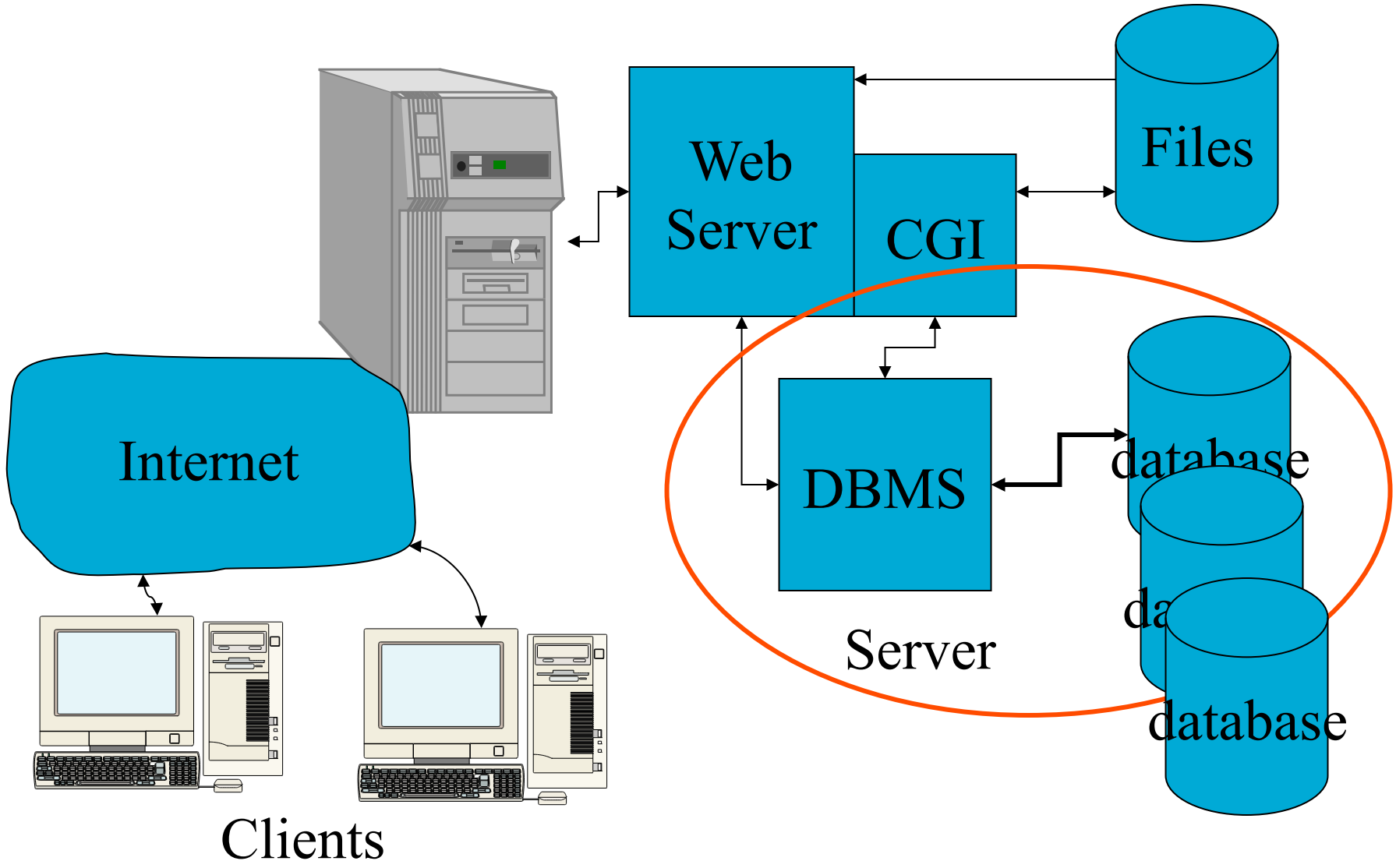
Transaction Processing

- Transactions should be ACID:
  - Atomic – Results of transaction are either all committed or all rolled back
  - Consistent – Data is transformed from one consistent state to another
  - Isolated – The results of a transaction are invisible to other transactions
  - Durable – Once committed the results of a transaction are permanent and survive system or media failures

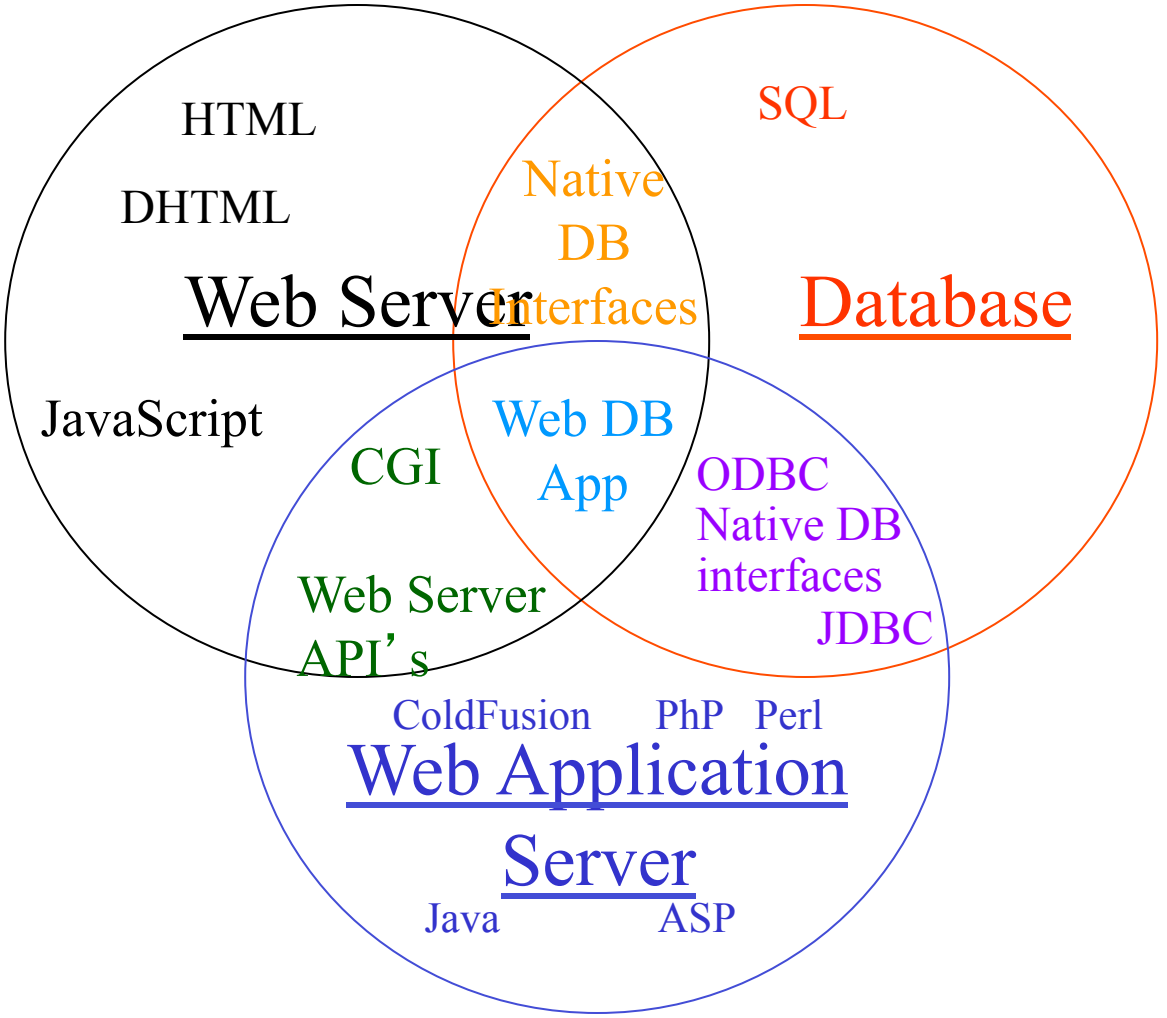
Why Use a Database System?

- Database systems have concentrated on providing solutions for all of these issues for scaling up Web applications
  - Performance
  - Scalability
  - Maintenance
  - Data Integrity
  - Transaction support
- While systems differ in their support, most offer some support for all of these.

Dynamic Web Applications 2



Server Interfaces



Adapted from
John P Ashenfelter,
Choosing a Database for Your Web Site

What Database systems are available?

- Choices depend on:
  - Size (current and projected) of the application
  - Hardware and OS Platforms to be used in the application
  - Features required
    - E.g.: SQL? Upgrade path? Full-text indexing? Attribute size limitations? Locking protocols? Direct Web Server access? Security?
  - Staff support for DBA, etc.
  - Programming support (or lack thereof)
  - Cost/complexity of administration
  - Budget

Desktop Database Systems

System (producer)	Platform	SQL	ODBC	Scaling	Price
Access (Microsoft)	Windows	Yes	Yes	SQL Server	~$200
FoxPro (Microsoft)	Windows, Mac	Yes	Yes	SQL Server	~$200
FileMaker (FileMaker)	Windows, Mac	No	No	FileMaker Server	~$200
Excel (Microsoft)	Windows, Mac	No	Yes	Convert to Acces	~$200
Files (owner)	Windows, Mac	No	No	Import into DB	?

- Individuals or very small enterprises can create DBMS-enabled Web applications relatively inexpensively
- Some systems will require an application server (such as ColdFusion) to provide the access path between the Web server and the DBMS

Pros and Cons of Database Options

- Desktop databases
  - usually simple to set up and administer
  - inexpensive
  - often will not scale to a very large number of users or very large database size
  - May lack locking management appropriate for multiuser access
  - Poor handling for full-text search
  - Well supported by application software (Coldfusion, PHP, etc.)

Enterprise Database Systems

System	Platform	SQL	ODBC	JDBC	Web?
SQL-Server (Microsoft)	WIndowsNT -2000	Yes	Yes	?	Yes (IIS)
Oracle Internet Platform	Unix, Linux, NT	Yes	Yes	Yes	Yes
Informix Internet Foundation.2000	Unix, Linux, NT	Yes	Yes	Yes	Yes
Sybase Adaptive Server	Unix, Linux, NT	Yes	Yes	Yes	Yes
DB2 (IBM)	IBM Unix, Linux, NT	Yes	Yes	Yes	Yes?

- Enterprise servers are powerful and available in many different configurations
- They also tend to be VERY expensive
- Pricing is usually based on users, or CPU's

Pros and Cons of Database Options

- Enterprise databases
  - Can be very complex to set up and administer
    - Oracle, for example recommends RAID-1 with 7x2 disk configuration as a bare minimum, more recommended
  - Expensive
  - Will scale to a very large number of users
  - Will scale to very large databases
  - Incorporate good transaction control and lock management
  - Native handling of Text search is poor, but most DBMS have add-on text search options
  - Support for applications software (ColdFusion, PHP, etc.)

Free Database Servers

System	Platform	SQL	ODBC	JDBC	Web?
mSQL	Unix, Linux	Yes	Yes	No(?)	No?
MySQL	Unix, Linux, NT	Yes	Yes	No(?)	No?
PostgreSQL	Unix, Linux, NT	Yes	Yes	Yes	No?

- System is free, but there is also no help line.
- Include many of the features of Enterprise systems, but tend to be lighter weight
- Versions may vary in support for different systems
- Open Source -- So programmers can add features

Pros and Cons of Database Options

- Free databases
  - Can be complex to set up and administer
  - Inexpensive (FREE!)
  - usually will scale to a large number of users
  - Incorporate good transaction control and lock management
  - Native handling of Text search has improved, and there are IR-like capabilities in MySQL and PostgreSQL
  - Support for applications software (ColdFusion, PHP, etc.)

Embedded Database Servers

System	Platform	SQL	ODBC	JDBC	Web?
Oracle Berkeley DB	Unix, Linux, Win	No	No	Java API	No?
Solid	Unix, Linux, Win	Yes	Yes	Yes	Yes
SQLite	Unix,Linux,Win	Yes	No	No	Yes

- May require programming experience to install
- Tend to be fast and economical in space requirements
- Includes many NOSQL databases

Pros and Cons of Database Options

- Embedded databases
  - Must be embedded in a program
  - Can be incorporated in a scripting language
  - inexpensive (for non-commercial application)
  - May not scale to a very large number of users (depends on how it is used)
  - (May) Incorporate good transaction control and lock management
  - Text search support is minimal
  - May not support SQL

NOSQL Databases

System	Platform	SQL	ODBC	JDBC	Web?
MongoDB	Unix, Linux, Win	No	No	No	?
REDIS	Unix, Linux, Win	NO	NO	No	?

Evaluation Criteria	Tokyo Cabinet + Tokyo Tyrant	Berkeley DB + MemcacheDB	Voldemort + BDB JE	Redis	MongoDB
Insertion (small data set)	4	2	2	4	4
Insertion (large data set)	1	1	1	1	3
Random Read (small data set)	4	2	2	4	4
Random Read (large data set)	2	2	1	3	3
Speed Consistence	1	3	2	4	4
Storage Efficiency	4	2	1	2	3
Horizontal Scalability	4	3	2	3	4
Manageability	4	4	2	2	4
Stability	4	4	2	3	4
Feature Set	1	1	1	1	2
Project Activeness and Community Support	2	1	1	2	4

PerfectMarket Content Matters

Database Security

- Different systems vary in security support:
  - Views or restricted subschemas
  - Authorization rules to identify users and the actions they can perform
  - User-defined procedures (and rule systems) to define additional constraints or limitations in using the database
  - Encryption to encode sensitive data
  - Authentication schemes to positively identify a person attempting to gain access to the database

Views

- A subset of the database presented to some set of users.
  - SQL: CREATE VIEW viewname AS SELECT field1, field2, field3,…, FROM table1, table2 WHERE <where clause>;
  - Note: “queries” in Access function as views.

Authorization Rules

- Most current DBMS permit the DBA to define “access permissions” on a table by table basis (at least) using the GRANT and REVOKE SQL commands.
- Some systems permit finer grained authorization (most use GRANT and REVOKE on variant views.
- Some desktop systems have poor authorization support.

Database Backup and Recovery

- Backup
- Journaling (audit trail)
- Checkpoint facility
- Recovery manager

Web Application Server Software

- ColdFusion
- PHP
- ASP
- JSP
- All of the are server-side scripting languages that embed code in HTML pages

Coldfusion

- Coldfusion was one of the first server-side scripting languages and it is still available and used
  - Originally produced by a company called Allaire, it is now owned by Adobe and is in version 11
  - It has always been a commercial product since the mid-1990's

What ColdFusion is Good for

- Putting up databases onto the Web
- Handling dynamic databases (Frequent updates, etc)
- Making databases searchable and updateable by users
- The basic scripting elements are simple, and similar in style to other server-side scripting languages (but the syntax is often different)

Coldfusion

- The Coldfusion engine runs in parallel with the web server, and is passed any page in the web server directories that has the appropriate file name extension (.cfm)
- The engine processes any Coldfusion script on the web page and passes back an HTML page with the scripts replaced by the script result
- As a simple example…

Coldfusion Templates

- Assume we have a database named contents_of_my_shopping_cart.mdb -- single table called contents...
  - With attributes “Item”, “Date_of_item”, “Price”
- Create an HTML page (uses extension .cfm), before <HEAD>...
- <CFQUERY NAME= ”cart" DATASOURCE=“contents_of_my_shopping_cart"> SELECT * FROM contents ; </CFQUERY>

Coldfusion Templates cont.

- <HTML>… the cfquery goes here…
- <HEAD>
- <TITLE>Contents of My Shopping Cart</TITLE>
- </HEAD>
- <BODY>
- <H1>Contents of My Shopping Cart</H1>
- <CFOUTPUT QUERY= ”cart">
- <B>#Item#</B>
- #Date_of_item#
- $#Price# <P>
- </CFOUTPUT>
- </BODY>
- </HTML>

Templates cont.

Contents of My Shopping Cart

Bouncy Ball with Psychedelic Markings
12 December 1998
$0.25

Shiny Blue Widget
14 December 1998
$2.53

Large Orange Widget
14 December 1998
$3.75

CFIF and CFELSE

```
<CFOUTPUT QUERY= ”cart">
        Item: #Item# <BR>
<CFIF #Picture# EQ"">
        <IMG SRC=“generic_picture.jpg"> <BR>
<CFELSE>
        <IMG SRC="#Picture#"> <BR>
</CFIF>
</CFOUTPUT>
```

More Templates

```
<CFQUERY DATASOURCE = "AZ2">
INSERT INTO Employees(firstname, lastname,
phoneext) VALUES( '#firstname#' , '#lastname#' ,
'#phoneext#' ) </CFQUERY>
<HTML><HEAD><TITLE>Employee Added</TITLE>
<BODY><H1>Employee Added</H1>
<CFOUTPUT>
Employee <B>#firstname# #lastname#</B> added.
</CFOUTPUT></BODY>
</HTML>
```

CFML ColdFusion Markup Language

- Read data from and update data to databases and tables
- Create dynamic data-driven pages
- Perform conditional processing
- Populate forms with live data
- Process form submissions
- Generate and retrieve email messages
- Perform HTTP and FTP function
- Perform credit card verification and authorization
- Read and write client-side cookies

Next time

- More on Database Applications: PHP and MySQL