

Connecting Databases to the
Outside World

Chapter 9

Class 08: Connecting Databases to the Outside
World

1

Deployment Models

- A *deployment model* describes the way that databases are connected with database users and other computer systems within the enterprise computing infrastructure.
- *Infrastructure* refers to the internal structure that organizes all the computing resources of an enterprise, including databases, applications, computer hardware, and the network.

Class 08: Connecting Databases to the Outside
World

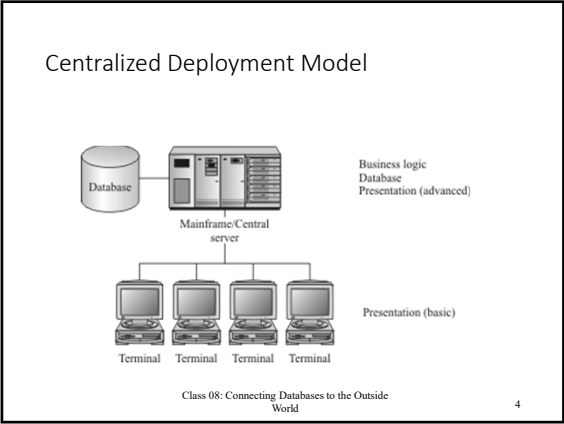
2

Deployment Models

- **Deployment Models:**
 - Centralized (Mainframe) Model
 - Distributed Model
- **Client/Server Models**
 - Two-Tier
 - Three-Tier
 - N-Tier (aka Internet Computing Model)
 - Cloud Computing Model

Class 08: Connecting Databases to the Outside
World

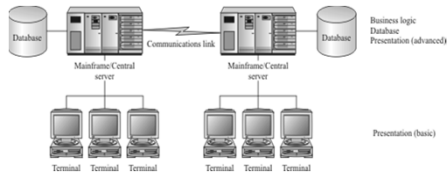
3



- ### Centralized Deployment Model
- **The original model for database connections**
 - **Database users equipped with “dumb terminals”**
 - Terminal presents screens, move the cursor on the screen, and capture keystrokes
 - **All other processing occurs on the mainframe or centralized server**
 - Business logic, database(s), advanced presentation (charts, graphs, colors)
- Class 08: Connecting Databases to the Outside World 5

- ### Centralized Deployment Model
- **Benefits:**
 - Very easy administration
 - Lower development costs (few specialists)
 - Potentially higher data input productivity
 - **Drawbacks:**
 - Single Point of Failure (the central server)
 - Primitive graphical displays
 - Terminal has a big desktop “footprint” for its function
- Class 08: Connecting Databases to the Outside World 6

Distributed Deployment Model



Class 08: Connecting Databases to the Outside World

7

Distributed Model

- *Distributed* means the database and/or application is divided into parts, which are placed on different computing devices
- Developed in the late 70s and early 80s
- *Transparent* to the users if done correctly
- Early deployments did not live up to the hype due to high costs and immature network technology

Class 08: Connecting Databases to the Outside World

8

Distributed Model

- **Benefits**
 - Improved fault tolerance; no single point of failure
 - Potential performance improvement by placing data and application logic closer to users
- **Drawbacks**
 - Much more complicated
 - Data synchronization can degrade performance
 - More expensive than Centralized Model
 - Lack of guidelines and standards for partitioning

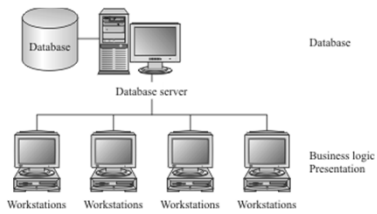
Class 08: Connecting Databases to the Outside World

9

Client/Server Model

- Involves one or more shared computers called *servers*
- Individual users have workstations called *clients*
- Arrived in the 1980s amid unprecedented marketing hype ("better, faster, cheaper")
- Evolved from 2-Tier to 3-Tier to N-Tier

Two-Tier Client/Server Model



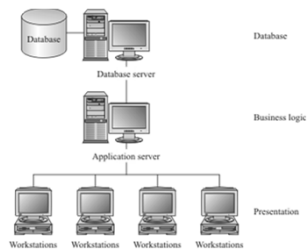
Two-Tier Client/Server Model

- Almost the opposite of the Centralized Model
- All business and presentation logic placed on the client workstation
- Only the database remains on the central server
- Most projects late and over budget – some monumental failures

Two-Tier Client/Server Model

- **Benefits**
 - Greatly improved user interface compared with dumb terminals
 - Potential for improved performance due to dedicated client workstation
- **Drawbacks**
 - Very expensive client workstations
 - Software updates an administrative nightmare
 - Development much more complicated and often more expensive

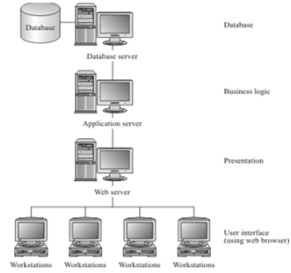
Three-Tier Client/Server Model



Three-Tier Client/Server Model

- Application logic moved to a shared server called an *Application Server*
- **Benefits**
 - Solved administrative issues with software upgrades
 - Improved scalability
 - Retained user interface advantages
 - Less expensive client workstations
- **Drawbacks**
 - Still more complicated than Centralized Model
 - Custom presentation methods and logic added to expense and limited portability

N-Tier Client/Server Model



Class 08: Connecting Databases to the Outside World

16

N-Tier Client/Server Model

- Also called the Internet Computing Model
- Presentation handled by web pages, managed by Web Server using industry- standard protocols
- Application and Database servers roughly the same as the Three-Tier Model
- Since client only runs a web browser, this is practically a return to the Centralized Model

Class 08: Connecting Databases to the Outside World

17

N-Tier Client/Server Model

- **Benefits**
 - Industry-standard presentation protocols
 - Same architecture for internet and intranet
 - Retains all the benefits of Three-Tier model
- **Drawbacks**
 - Security challenges (Internet and the Web not designed with security in mind)
 - Each layer requires a development specialist
 - Potentially more hardware, but perhaps not at higher overall cost

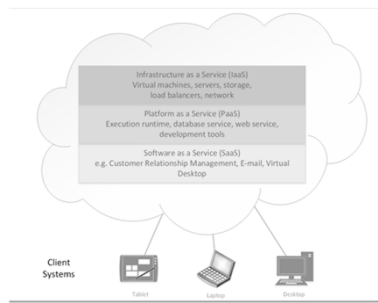
Class 08: Connecting Databases to the Outside World

18

Cloud Computing Model (added material)

- Developed for organizations that need:
 - The ability to rapidly provision and release computing resources based on demand
 - Minimize the need for significant management effort or day-to-day interaction with the service provider.
- Instead of a dedicated set of computing resources that an organization must acquire, install, and configure themselves, the computing resources appear as a shared pool of resources, including networks, servers, storage, applications, and other services.

Cloud Computing Model



Cloud Computing Resources

- Available resources usually organized as:
 - **Clients:** Devices can be anything from ultra-thin devices to fully configured desktop computers
 - Require only a browser and an internet connection
 - **Software as a Service (SaaS):** Contains application software that is used by the clients
 - Usually requires a monthly or annual fee per user
 - Example: Salesforce.com
 - **Platform as a Service (PaaS):** Provides cloud users with platforms and databases as a service
 - Examples: Microsoft Azure, Amazon Relational Database Service (RDS)

Cloud Computing Resources

• **Infrastructure as a Service (IaaS):** a virtual implementation of the physical hardware devices (that is, all the servers, networks, load balancers, storage, and system management services).

- Forms the equivalent of the stack of servers shown in the N-tier client-server model diagram earlier in this topic.
- Individuals and businesses pay a monthly or annual fee for running virtual servers, networks, and storage in the cloud.
- Cloud storage was one of the earliest such services, and is therefore highly available and heavily used. Examples include Amazon S3, Microsoft OneDrive, Google Drive, Box, Dropbox, and many more.

Class 08: Connecting Databases to the Outside World

22

Cloud Computing Advantages

- **Flexibility:** Services can be acquired and released quickly based on the current and expected processing loads.
- **No Capital Investment:** Businesses using the cloud computing platform are not burdened with capital investments for hardware and software purchases -- cloud service provider assumes the burden of these capital investments.
- **Automatic Software Updates:** Businesses using the cloud computing platform do not have to manage software updates, or be concerned with clients running different (outdated) versions of the application -- cloud vendor manages software updates automatically
- **Improved Mobility:** Business users can work from anywhere that has an Internet connection -- no need to take a high-powered computing device, or a local database with them.

Class 08: Connecting Databases to the Outside World

23

Cloud Computing Disadvantages

- **Downtime:** Cloud users are completely dependent on the availability of the cloud computing platform and a reliable network to connect them to it.
- **Security and Privacy Concerns:** Organizations handling sensitive or regulated data must take steps to ensure that an intrusion into their cloud storage does not put their business at risk.
 - While data is probably safer stored in the cloud than on laptops and tablets that can be lost or stolen, a successful intrusion of a cloud storage platform can have enterprise-wide consequences.
 - For example, the Amazon Web Services EC2 Console for a company named Code Space was compromised, and when Code Space started locking down resources instead of paying the hacker's ransom demand, the hacker deleted all the data, including all of Code Space's S3 backups. Code Space had little choice but to discontinue operations and go out of business.

Class 08: Connecting Databases to the Outside World

24

Cloud Computing Disadvantages

- **Vulnerability to Attack:** Every component is potentially accessible via the Internet.
 - Clients must take steps to identify potential threats and mitigate them as much as possible.
 - Routine monitoring for unusual activity can also mitigate risk.
- **Cloud Computing Cost:** Unlike your internal data center, someone is making a profit by selling cloud computing access to your business users.
 - Can make cloud computing more expensive under some circumstances, particularly for small or short-term projects.
 - Cost reduction efforts can help here.

Class 08: Connecting Databases to the Outside World

25

Introduction to the Internet and the Web

- **Internet:** A world-wide collection of interconnected computer networks
 - Began in late '60s and early '70's as US DOD ARPANET
 - TCP/IP adopted as standard in 1982
 - Other protocols include FTP, SMTP, Telnet, DNS, POP, etc.

Class 08: Connecting Databases to the Outside World

26

Introduction to the Internet and the Web

- **Intranet:** A web site or group of sites belonging to an organization, accessible only by the members of the organization.
 - Resides behind a firewall
- **Extranet:** An intranet that is partially accessible to authorized outsiders.

Class 08: Connecting Databases to the Outside World

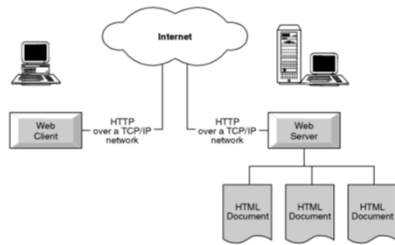
27

The Web

• **World Wide Web:** A hypermedia-based system that provides a simple “point and click” means of browsing information on the Internet using hyperlinks.

- Hyperlinks allow users to navigate pages in a non-sequential manner
- Clients use web browser to present pages
- Web server hosts pages and responds to client requests
- Pages can be static or dynamic

Basic Components of Web Environment



HyperText Transfer Protocol (HTTP)

- **HTTP:** The protocol used to transfer web pages through the Internet.
 - HTTP is a request based paradigm
 - “Stateless” protocol – each connection closed once the server provides a response
 - Statelessness makes it difficult to support the concept of a session, which is essential to basic DBMS transactions.

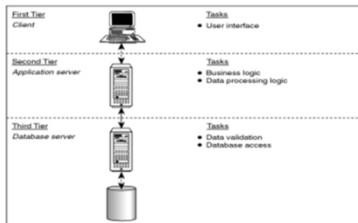
HyperText Markup Language (HTML)

- **HTML**: The document formatting language used to design most Web pages.
 - System for marking up or tagging a document for publication on the Web.
 - Derived from the Standardized General Markup Language (SGML), ISO 1986 standard.

Uniform Resource Locators (URLs)

- **URL**: A string of alphanumeric characters that represents the location or address of a resource on the Internet and how that resource should be accessed.
 - Defines uniquely where resource can be found.
 - Ultimately translates to an IP Address, port and a protocol (HTTP, FTP, etc.):
<protocol>://<host>[:<port>]/absolute path [?arguments]

Typical Three-Tier Architecture (Web Server not shown)



Advantages of Web-DBMS Approach

- DBMS advantages (data driven web pages)
- Simplicity
- Platform independence
- Graphical User Interface (GUI)
- Standardization
- Cross-platform support
- Transparent network access
- Scalable deployment
- Innovation

Class 08: Connecting Databases to the Outside World 34

Challenges of Web-DBMS Approach

- Security (must buy/build your own)
- Scalability (unpredictable peak loads)
- Limited functionality of HTML (partially offset by XML, JSON and others)
- Statelessness
- Bandwidth
- Performance

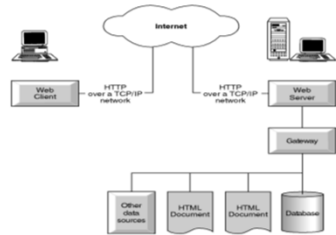
Class 08: Connecting Databases to the Outside World 35

Common Gateway Interface (CGI)

- **CGI:** A specification for transferring information between a Web server and a CGI program
 - Defines how scripts communicate with Web servers.
 - URL points to script; Server launches it
 - Script written in a variety of languages

Class 08: Connecting Databases to the Outside World 36

Common Gateway Interface (CGI)



Class 08: Connecting Databases to the Outside World

37

Common Gateway Interface (CGI)

• Advantages

- Simplicity
- Language and Web server independence
- Wide acceptance

• Disadvantages

- Web server always between client and data base
- No transaction support (stateless)
- Not intended for long exchanges
- New process (thread) for each CGI script launched
- Security issues

Class 08: Connecting Databases to the Outside World

38

Server Side Includes

- Server parses document before sending it to the browser
- SSI commands embedded in document cause server to execute a program (like CGI) and incorporate the output into the document

Class 08: Connecting Databases to the Outside World

39

HTTP Cookies

- **Cookies:** small text files stored on the client
- CGI script creates cookie and has Web server send it to the client for storage
- Later, CGI script requests cookie, and if browser can find it, it is returned to the script
- Not all browsers support cookies; not all users allow them

Extending the Web Server

- **Non-CGI Gateways:** Netscape Server API, Microsoft Internet Information Server (IIS) API
- **Advantages:** Performance (over CGI), additional features and functions, runs in Server address space (efficient)
- **Disadvantages:** Proprietary, typically much more complex

Open Database Connectivity (ODBC)

- Standard API for connecting programs to DBMSs.
- Based on a CLI (Call Level Interface) first defined for Microsoft Access in 9/92.
- While Microsoft offered the first commercial release, it is not a Microsoft standard.
- Independent of language and platform – applications can be ported by changing the driver.
- Most commercial DBMSs and most commercial applications support ODBC.

Java

- Proprietary language developed by Sun Microsystems (later JavaSoft division; now Oracle)
- Became a wildly popular standard programming language for Web computing
- A type-safe, object programming language that can be used to build client components (**applets**) and server components (**servlets**)
- Machine independent target architecture

Class 08: Connecting Databases to the Outside World 43

JDBC

- Modeled after Open Data Base Connectivity (ODBC)
- An API for Java that supports a wide range of relational DBMS products
 - Embedded SQL for Java: SQL passed as string to Java methods; Embedded SQL Processor translates Java/SQL to JDBC calls
 - Direct Mapping of RDBMS tables to Java Classes: planned by many; programmers operate on objects with SQL automatically generated; each table implemented as a class

Class 08: Connecting Databases to the Outside World 44

JDBC SQL Conformance

- Relational DBMSs do not support advanced functions in the same way
- JDBC API designed to support the various dialects of SQL
- JDBC also provides ODBC-style escape clauses to pass proprietary statement portions directly to the DBMS

Class 08: Connecting Databases to the Outside World 45

Other Java Connection Methods

- JSQL: Java with embedded SQL, an extension of ISO/ANSI standard for SQL embedded in other host languages
- JRB (Java Relational Binding): Middleware product from O2 Technology that bridges Java to Relational DBMSs

Class 08: Connecting Databases to the Outside World 46

Middleware Solutions

- *Middleware* is software that mediates the differences between an application program and the services available on the network, or between two disparate applications.
- Middleware can make an RDBMS look like an OO DBMS, hiding all the relational implementation details from the applications.

Class 08: Connecting Databases to the Outside World 47

Scripting Languages

- Script code is embedded in HTML and downloaded every time the page is accessed
- Two popular ones: JavaScript and VBScript
- Allow creation of functions embedded in HTML as well as generating HTML "on the fly"
- Interpretive (not compiled)

Class 08: Connecting Databases to the Outside World 48

JavaScript and JScript

- Virtually identical interpreted scripting languages from Netscape and Microsoft
- Object-based scripting language
- Resembles Java, but without static typing and strong type checking
- Replaces or extends CGI for database access

Class 08: Connecting Databases to the Outside World 49

VBScript

- Microsoft proprietary interpreted scripting language with syntax like Visual Basic
- Procedural language with subroutines as basic programming unit
- For security, no functions interact directly with files on the client system

Class 08: Connecting Databases to the Outside World 50

Security

- TCP/IP and HTML not designed with security in mind.
- All internet traffic travels "in the clear"
- Information transmitted to the client may have executable content that could perform malicious actions

Class 08: Connecting Databases to the Outside World 51

Proxy Servers

- Computer that sits between Web browser and Web server
 - Can improve performance by caching results
 - Can filter requests
 - Can hide network residing behind it

Class 08: Connecting Databases to the Outside World 52

Firewalls

- Hardware/software/both designed to prevent unauthorized access to/from a private network
 - Packet Filtering: inspect packets entering/leaving network
 - Application Gateway: applies security to services like FTP and Telnet
 - Circuit-level Gateway: security mechanisms for TCP or UDP (Universal Datagram Protocol)
 - Proxy Server: hides true network addresses

Class 08: Connecting Databases to the Outside World 53

SSL and S-HTTP

- Secure Sockets Layer (SSL):
 - Encryption protocol developed by Netscape
 - Uses private key to encrypt data over SSL connection
 - Designed to prevent eavesdropping, tampering and message forgery
- Secure HTTP (S-HTTP, HTTPS):
 - Modified HTTP that establishes secure connection between client and server
 - Client and server authenticate each other

Class 08: Connecting Databases to the Outside World 54

Java Security

- "Sandbox" ensures that an untrusted application cannot gain access to system resources
- Class Loader: checks that class is correct format and that application/applet does not violate system security
- Bytecode Verifier: verifies code before JVM allows it to run

Class 08: Connecting Databases to the Outside World 55

Extensible Markup Language (XML)

- Allows designers to create their own tags (markup language extended based on need)
- Ability to define database schema
- Relative Objects and References: allows reference by position of item in a list
- Support for Bi-Directional Links: documents reference each other

Class 08: Connecting Databases to the Outside World 56
