

Database Management System – 2 (Database users, Advantages of using DBMS, History of DBMS)

Ajay James

Asst. Prof in CSE

Government Engineering College Thrissur

Database Users

1. Actors on the Scene

- Who use and control the database content, and those who design, develop and maintain database applications
- **Database Administrators**
- **Database Designers**
- **End-users**
- **System Analysts and Application Programmers**

2. Workers behind the scene

- who design and develop the DBMS software and related tools
- DBMS system designers and implementers, Tool developers, Operators and maintenance personnel

Database Administrators

- Chief administrator to oversee and manage these resources
- **Database administrator (DBA)**
- Authorizing access to the database
- Coordinating and monitoring its use
- Acquiring software and hardware resources
- Accountable for problems such as security breaches and poor system response time

Database Designers

- Identifying the data to be stored in the database
- Choosing appropriate structures to represent and store this data
- Understand their requirements of all users
- Create a design that meets these requirements
- Interact with each potential group of users and develop views
- Final design - Supporting the requirements of all user groups

End Users – 4 types

- People whose jobs require access to the database for querying, updating, and generating reports
- **Casual end users**
 - occasionally access the database, but they may need different information each time
 - Use a sophisticated database query language to specify their requests
 - middle- or high-level managers or other occasional browsers
- **Naive or parametric end users**
 - constantly querying and updating the database, using standard types of queries and updates—called canned transactions
 - Bank tellers, Reservation agents for airlines, hotels, and car rental companies

End Users

- **Sophisticated end users**
 - include **engineers, scientists, business analysts**
 - thoroughly familiarize themselves with the facilities of the DBMS
 - Use it to implement their own applications to meet their complex requirements
- **Standalone users**
 - maintain personal databases by using ready-made program packages
 - Example - tax package

System Analysts and Application Programmers

- **System analysts**
 - determine the requirements of end users
 - develop specifications for standard canned transactions
- **Application programmers**
 - implement these specifications as programs
- Analysts and programmers—referred to as **software developers or software engineers**

Workers behind the Scene

- Design, development, and operation of the DBMS software and system Environment
- **DBMS system designers and implementers**
 - design and implement the DBMS modules and interfaces as a software package
 - modules for implementing the catalog, query language processing, interface processing, accessing and buffering data, controlling concurrency, and handling data recovery and security
- **Tool developers**
 - design and implement tools (optional)
 - packages for database design, performance monitoring, natural language or graphical interfaces, prototyping, simulation, and test data generation.
- **Operators and maintenance personnel (system administration personnel)**
 - responsible for the actual running and maintenance of the hardware and software environment for the database system

Advantages of Using the DBMS Approach

1. Controlling redundancy
2. Restricting unauthorized access to data.
3. Providing persistent storage for program Objects
4. Providing Storage Structures for efficient Query Processing
5. Providing backup and recovery services
6. Providing multiple interfaces to different classes of users.
7. Representing complex relationships among data
8. Enforcing integrity constraints on the database
9. Permitting Inferencing and Actions Using Rules
10. *Additional Implications*

Advantages

1. Controlling Redundancy

- Storing the same data multiple times leads to several problems
- *Duplication of effort*
- *Storage space is wasted* when the same data is stored repeatedly
- Files that represent the same data may become *inconsistent*

2. Restricting Unauthorized Access

- Only authorized persons are allowed to access
- Some users may only be permitted to retrieve data, whereas others are allowed to retrieve and update
- Security and authorization subsystem

Advantages

3. Providing Persistent Storage for Program Objects

- Provide persistent storage for program objects and data structures.
- Object-oriented database systems

4. Providing Storage Structures and Search Techniques for Efficient Query Processing

- capabilities for *efficiently executing queries and updates*
- buffering or caching module

Advantages

5. Providing Backup and Recovery

- facilities for recovering from hardware or software failures
- Disk backup

6. Providing Multiple User Interfaces

- provide a variety of user interfaces
- Query languages for casual users, programming language interfaces for application programmers, forms and command codes for parametric users, and menu-driven interfaces and natural language interfaces for standalone users

Advantages

7. Representing Complex Relationships among Data

8. Enforcing Integrity Constraints

- Capabilities for defining and enforcing integrity constraints

9. Permitting Inferencing and Actions Using Rules

- Capabilities for defining deduction rules for inferencing new information from the stored database facts

Additional Implications of Using the Database Approach

- Potential for enforcing standards
- Reduced application development time
- Flexibility to change data structures
- Availability of up-to-date information
- Economies of scale

History of Database Applications

- Early Database Applications:
 - Hierarchical and Network Models were introduced in mid 1960s and dominated during the seventies
- Relational Model based Systems:
 - Relational model was originally introduced in 1970
 - Relational DBMS Products emerged in the early 1980s
- Object-oriented and emerging applications:
 - Object-Oriented Database Management Systems (OODBMSs) were introduced in late 1980s and early 1990s
 - to cater to the need of complex data processing in CAD and other applications.
 - *object-relational* DBMSs (ORDBMSs)
 - *Extended relational* systems add further capabilities (e.g. for multimedia data, XML, and other data types)

History of Database Applications

- Data on the Web and E-commerce Applications:
 - Web contains data in HTML (Hypertext markup language) with links among pages
 - New set of applications and E-commerce is using new standards like XML (eXtensible Markup Language)
 - Script programming languages such as PHP and JavaScript

History of Database Applications

- New functionality is being added to DBMSs in the following areas:
 - Scientific Applications
 - XML (eXtensible Markup Language)
 - Image Storage and Management
 - Audio and Video Data Management
 - Data Warehousing and Data Mining
 - Spatial Data Management
 - Time Series and Historical Data Management
- Emergence of Big Data Storage Systems and NOSQL Databases

When not to use a DBMS

- Costs of using a DBMS are due to the following:
 - High initial investment in hardware, software, and training
 - Overhead for providing security, concurrency control, recovery, and integrity functions
- Customized database applications
 - Simple, well-defined database applications that are not expected to change at all
 - Stringent, real-time requirements for some application programs
 - Embedded systems with limited storage capacity
 - No multiple-user access to data

Reference

- Elmasri R. and S. Navathe, Database Systems: Models, Languages, Design and Application Programming, Pearson Education 6th edition and 7th edition

Thank you