
Database management System
Security
Ctrl

Database is a collection of related data and collection of facts and figures that can be processed to produce information.

Mostly data represents recordable facts. Data in producing information, which is based on facts.
For example, if we have data about marks obtained by all students, we can then conclude about toppers and average marks.

A **database management system** stores data in such a way that it becomes easier , manipulate, and produce information.

What is Database

The database is a collection of inter-related data which is used to insert and delete the data efficiently. It is also used to organize the data in the form of a table, schema, views, and reports, etc.

- **For example:** The college Database organizes the data about the admin, staff, students and faculty etc.
- Using the database, you can easily retrieve, insert, and delete the information.

Database Management System (DBMS)

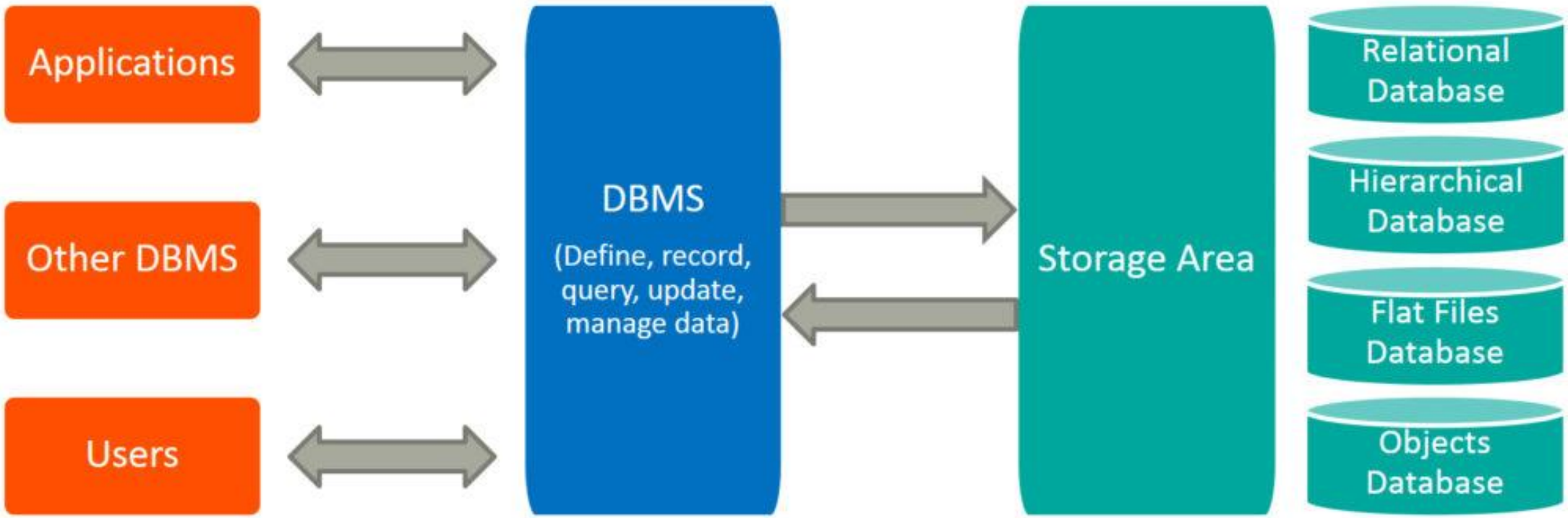
- Database management system is a software which is used to manage the database.

For example: MySQL, Oracle, etc are a very popular commercial database which is used in different applications.

- DBMS provides an interface to perform various operations like database creation, storing data in it, updating data, creating a table in the database and a lot more.

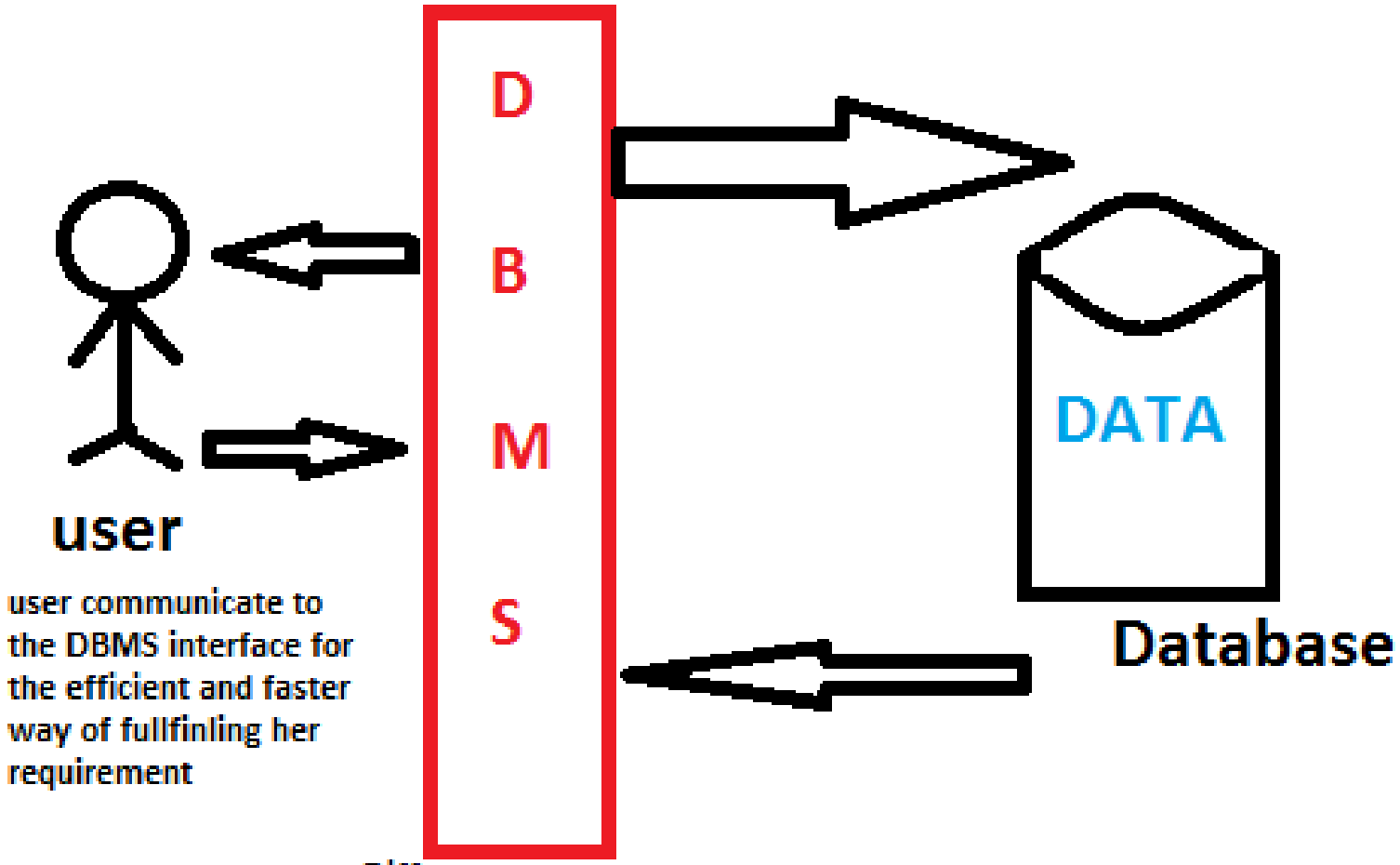
- It provides protection and security to the database. In the case of multiple users, it also maintains data consistency.

Database Management System



DBMS allows users the following tasks:

- **Data Definition:** It is used for creation, modification, and removal of definition that defines the organization of data in the database.
- **Data Updation:** It is used for the insertion, modification, and deletion of the actual data in the database.
- **Data Retrieval:** It is used to retrieve the data from the database which can be used by applications for various purposes.
- **User Administration:** It is used for registering and monitoring users, maintain data integrity, enforcing data security, dealing with concurrency control, monitoring performance and recovering information corrupted by unexpected failure.



user communicate to the DBMS interface for the efficient and faster way of fullfinling her requirement

Different programs to insert , delete , update and query database for information

Characteristics

. A modern DBMS has the following characteristics −

- **Real-world entity** − A modern DBMS is more realistic and uses real-world entities to design its architecture. It uses the behavior and attributes too. For example, a school database may use students as an entity and their age as an attribute.
- **Relation-based tables** − DBMS allows entities and relations among them to form tables. A user can understand the architecture of a database just by looking at the table names.
- **Isolation of data and application** − A database system is entirely different than its data. A database is an active entity, whereas data is said to be passive, on which the database works and organizes.

- **Less useful** − DBMS follows the rules of normalization, which splits a relation when any of its attributes is having redundancy in values. Normalization is a mathematically rich and scientific process that reduces data redundancy.
- **Consistency** − Consistency is a state where every relation in a database remains consistent. There exist methods and techniques, which can detect attempt of leaving database in inconsistent state. A DBMS can provide greater consistency as compared to earlier forms of data storing applications like file-processing systems.
- **Query Language** − DBMS is equipped with query language, which makes it more efficient to retrieve and manipulate data. A user can apply as many and as different filtering options as required to retrieve a set of data. Traditionally it was not possible where file-processing system was used.

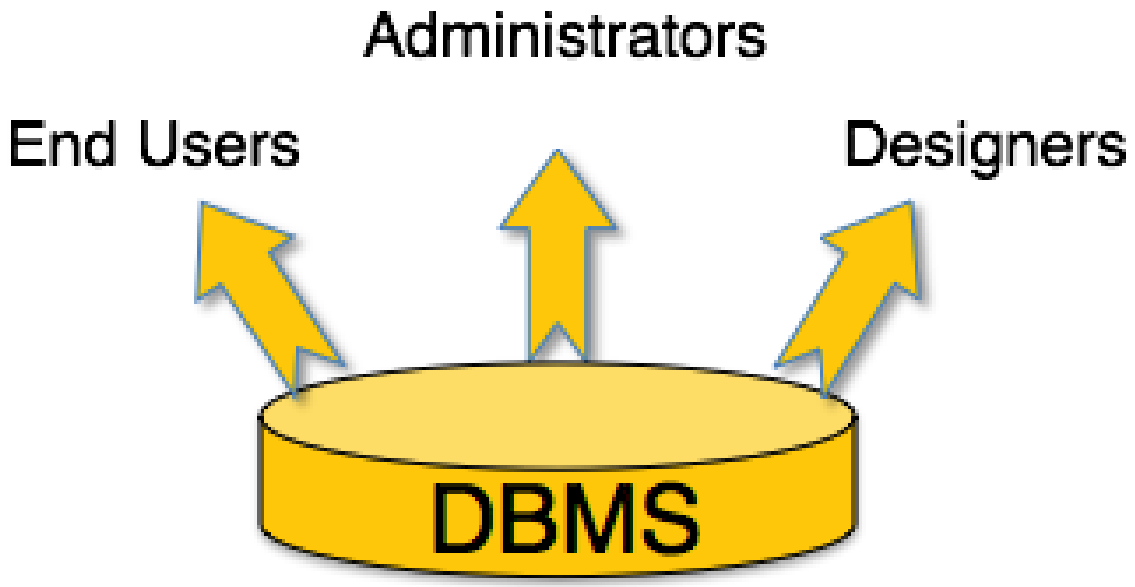
- **Multiuser and Concurrent Access** – DBMS supports multi-user environment and allows them to access and manipulate data in parallel. Though there are restrictions on transactions when users attempt to handle the same data item, but users are always unaware of them.
- **Multiple views** – DBMS offers multiple views for different users. A user who is in the Sales department will have a different view of database than a person working in the Production department. This feature enables the users to have a concentrate view of the database according to their requirements.

- **Security** –

Features like multiple views offer security to some extent where users are unable to access data of other users and departments. DBMS offers methods to impose constraints while entering data into the database and retrieving the same at a later stage. DBMS offers many different levels of security features, which enables multiple users to have different views with different features

Users

A typical DBMS has users with different rights and permissions who use it for different purposes. Some users find the store data and some back up it . The users of a DBMS can be broadly categorized as follows –



- **Administrators** − Administrators maintain the DBMS and are responsible for administrating the database. They are responsible to look after its usage and by whom it should be used. They create access profiles for users and apply limitations to maintain isolation and force security. Administrators also look after DBMS resources like system license, required tools, and other software and hardware related maintenance.

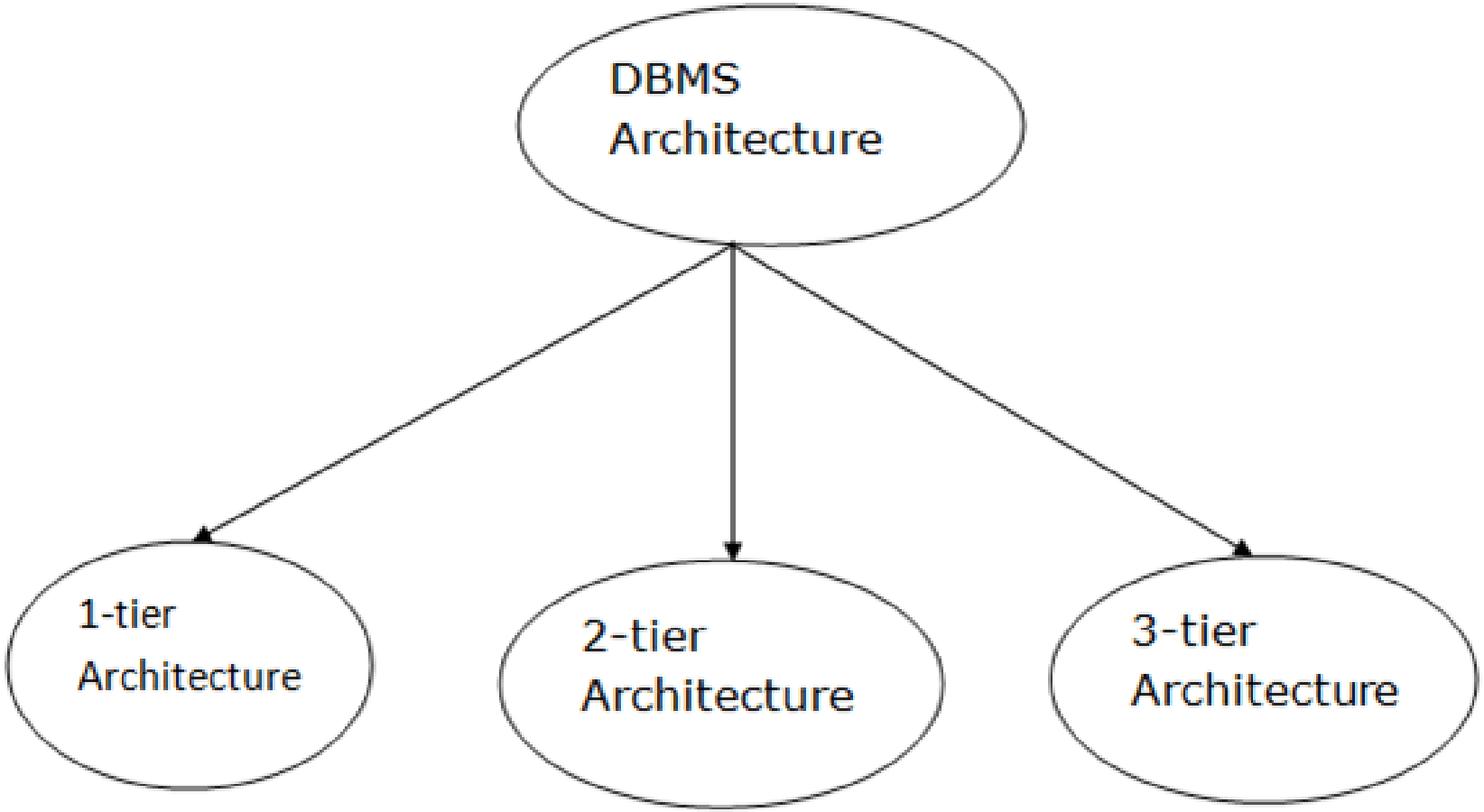
- **Designers** − Designers are the group of people who actually work on the designing part of the database. They keep a close watch on what data should be kept and in what format. They identify and design the whole set of entities, relations and views.

- **End Users** − End users are those who actually get the benefits of having a DBMS. End users can range from simple viewers who pay attention to the rates of users such as business analysis.

DBMS Architecture

A Database Management system is not always directly available for users and applications to access and store data in it.

A Database Management system can be **centralized** (all the data stored at one location), **decentralized** (multiple copies of database at different locations) depending upon its architecture.


DBMS
Architecture
1-tier
Architecture
2-tier
Architecture
3-tier
Architecture

1-tier DBMS

When the database is directly available to the user for using it to store data. Generally such a setup is used for local application development, where programmers communicate directly with the database for quick response.

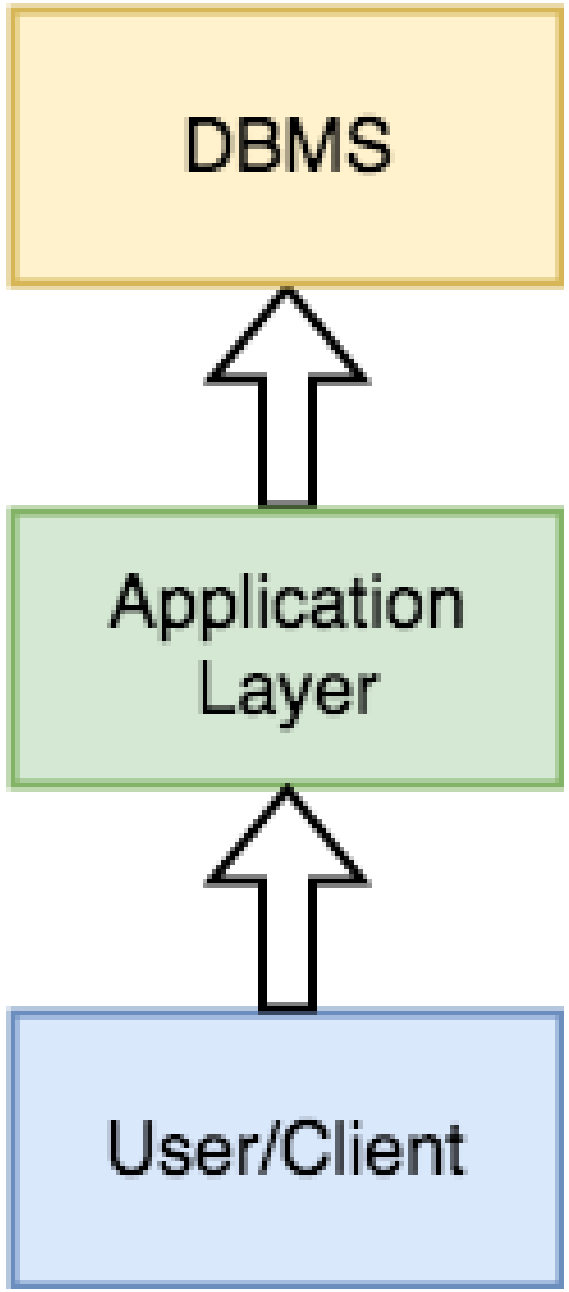
Database Architecture is logically of two types:

- 2-tier DBMS architecture
- 3-tier DBMS architecture

2-tier DBMS Architecture

2-tier DBMS architecture includes an **Application layer** between the user and the DBMS, which is responsible to communicate the user's request to the database management system and then send the response from the DBMS to the user.

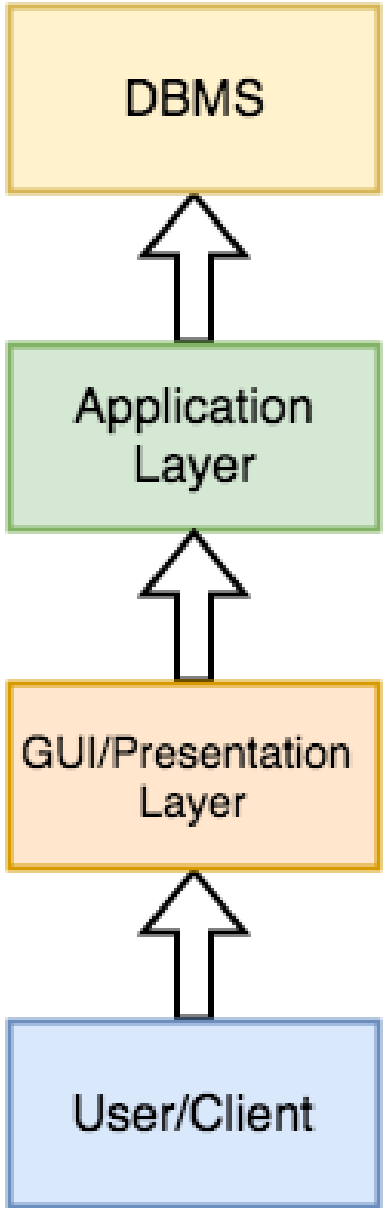
- An application interface known as **ODBC**(Open Database Connectivity) provides an API that allow client side program to call the DBMS. Most DBMS vendors provide ODBC drivers for their DBMS.

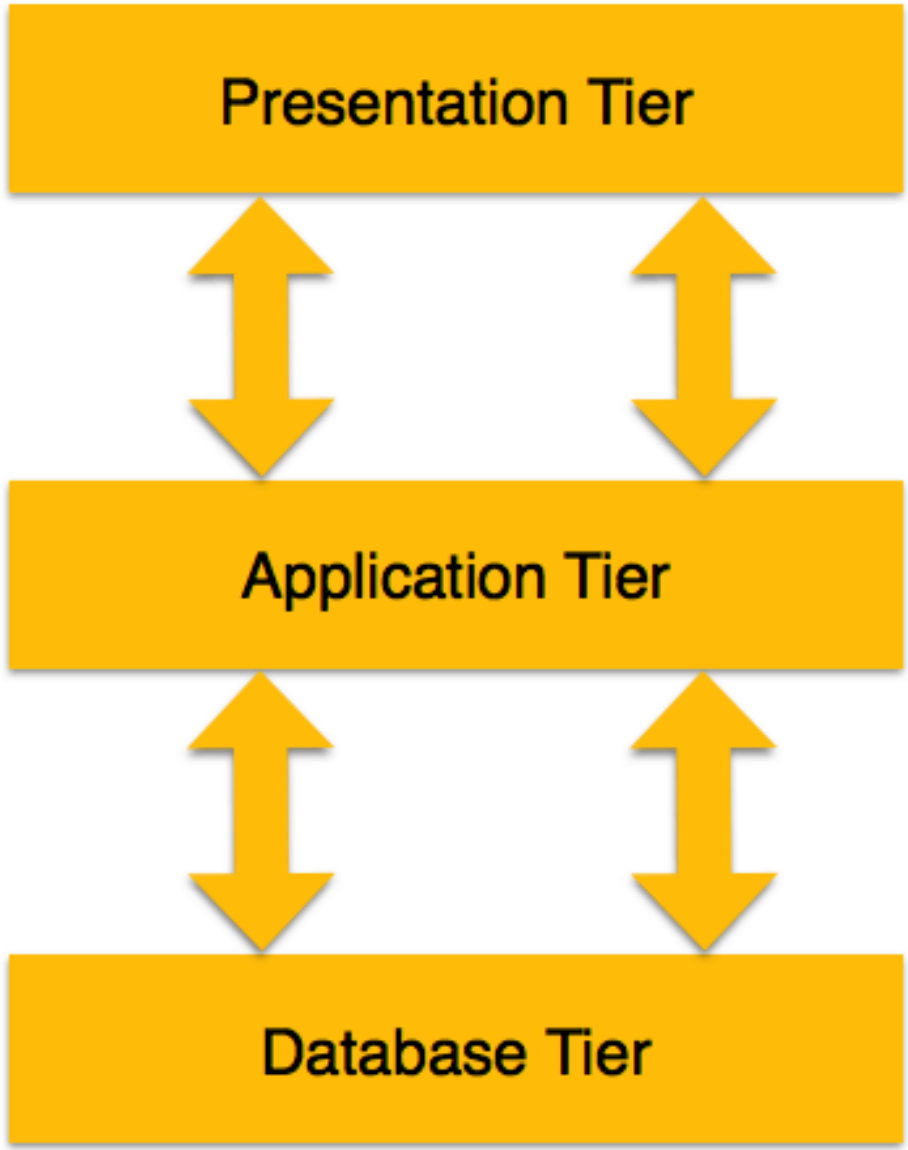

DBMS
Application Layer
User/Client

DBMS Architecture

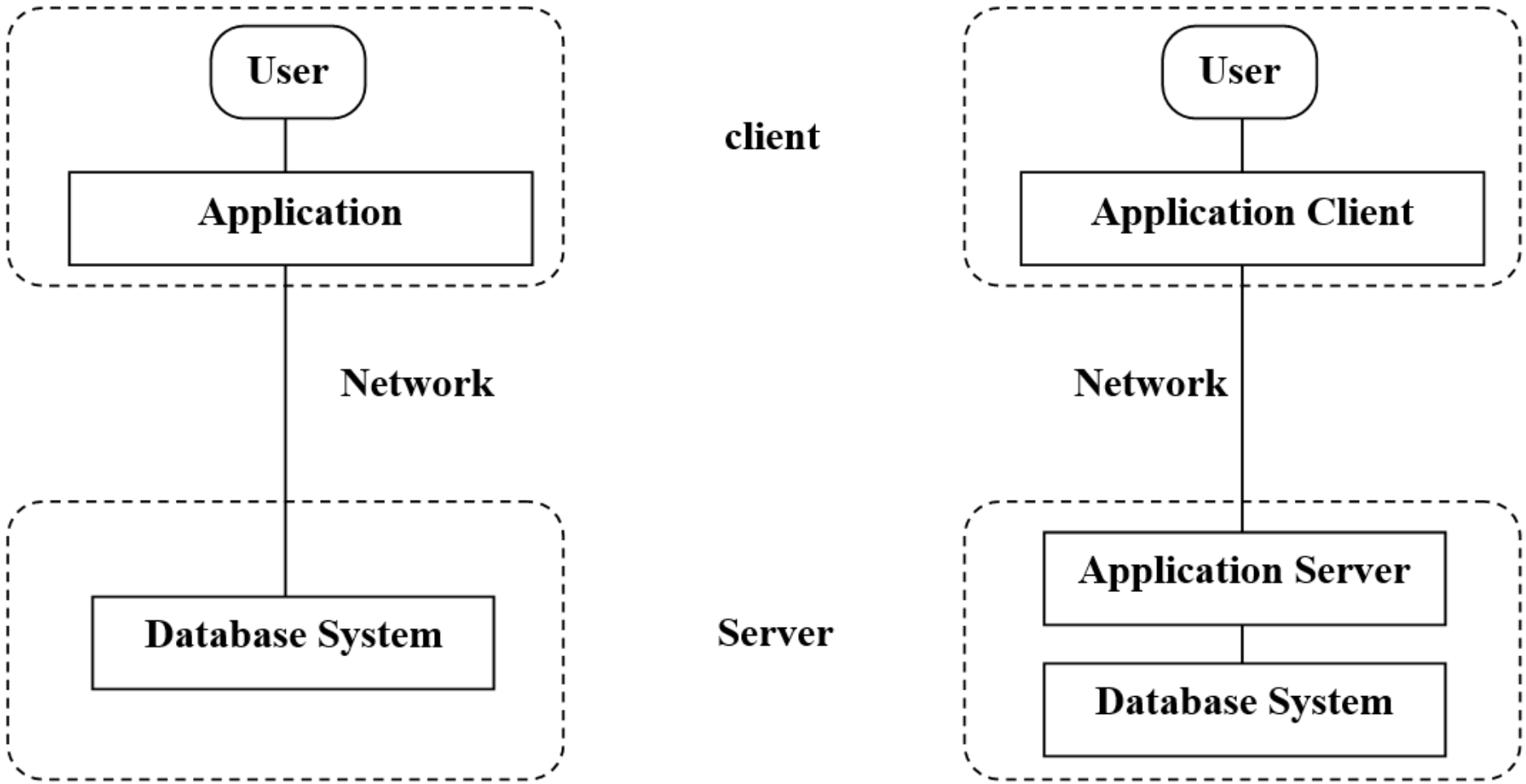
3-tier Architecture

- A 3-tier architecture separates its tiers from each other based on the complexity of the users and how they use the data present in the database.
- It is the most widely used architecture to design a DBMS.
- 3-tier DBMS architecture is the most commonly used architecture for web applications.


DBMS
Application Layer
GUI/Presentation Layer
User/Client


Presentation Tier
Application Tier
Database Tier

- **Database (Data) Tier** − At this tier, the database have along with its query processing languages. We also have the relations that define the data and their constraints at this level.
- **Application (Middle) Tier** − At this tier reside the application server and the programs that access the database. For a user, this application tier presents an abstracted view of the database. End-users are unaware of any existence of the database beyond the application. At the other end, the database tier is not aware of any other user beyond the application tier. Hence, the application layer sits in the middle and acts as a mediator between the end-user and the database.
- **User (Presentation) Tier** − End-users operate on this tier and they know nothing about any existence of the database beyond this layer. At this layer, multiple views of the database can be provided by the application. All views are generated by applications that reside in the application tier.

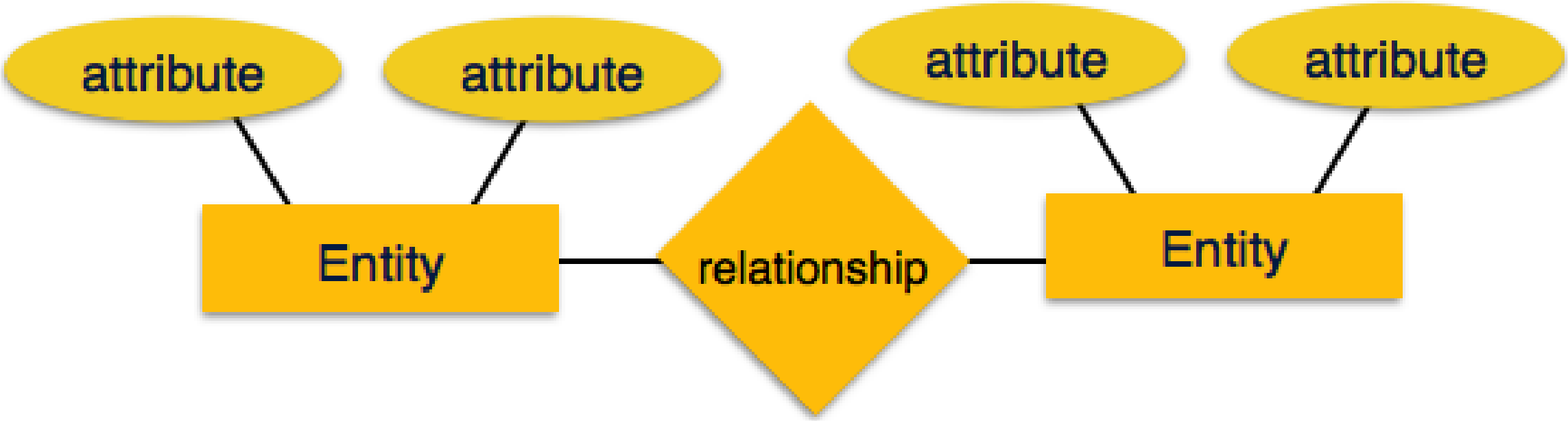


(a) Two-tier Architecture

(a) Three-tier Architecture

Entity-Relationship Model

- Entity-Relationship (ER) Model is based on the real-world entities and relationships among them. While formulating real-world scenario into the database model, the ER Model creates entity set, relationship set, and general attributes .
- ER Model is best used for the conceptual design of a database.
- ER Model is based on −
- **Entities** and their *attributes.*
- **Relationships** among entities.


attribute
attribute
attribute
attribute
Entity
relationship
Entity

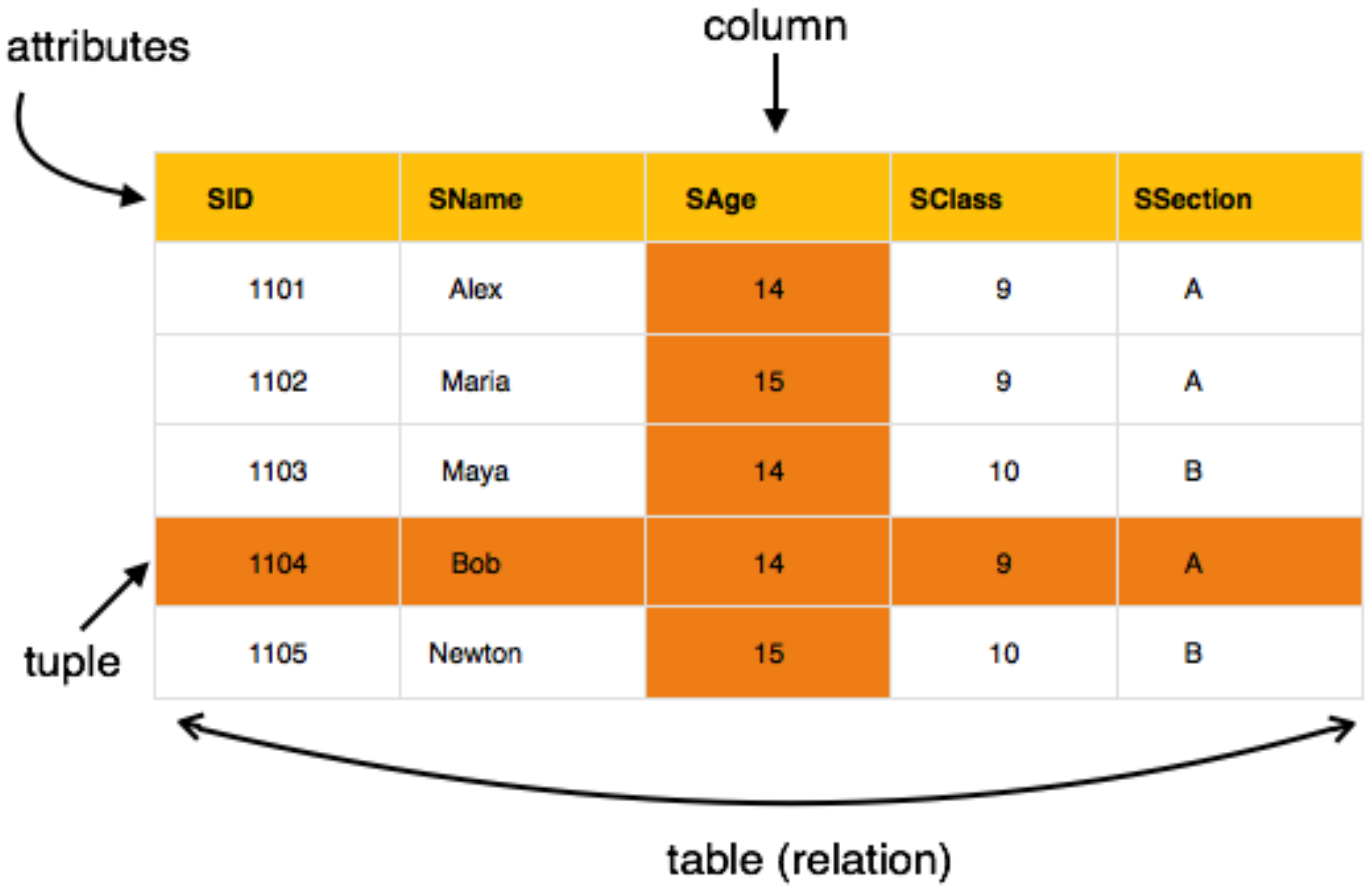
- **Entity** − An entity in an ER Model is a real-world entity having properties called **attributes**. Every **attribute** is defined by its set of values called **domain**.

For example, in a school database, a student is considered as an entity. Student has various attributes like name, age, class, etc.

- **Relationship** − The logical association among entities is called ***relationship***. Relationships are mapped with entities in various ways. Mapping cardinalities define the number of association between two entities.
- Mapping cardinalities −
  - one to one
  - one to many
  - many to one
  - many to many

Relational Model

The most popular data model in DBMS is the Relational Model. It is more scientific a model than others. This model is based on first-order predicate logic and defines a table as an **n-ary relation**.

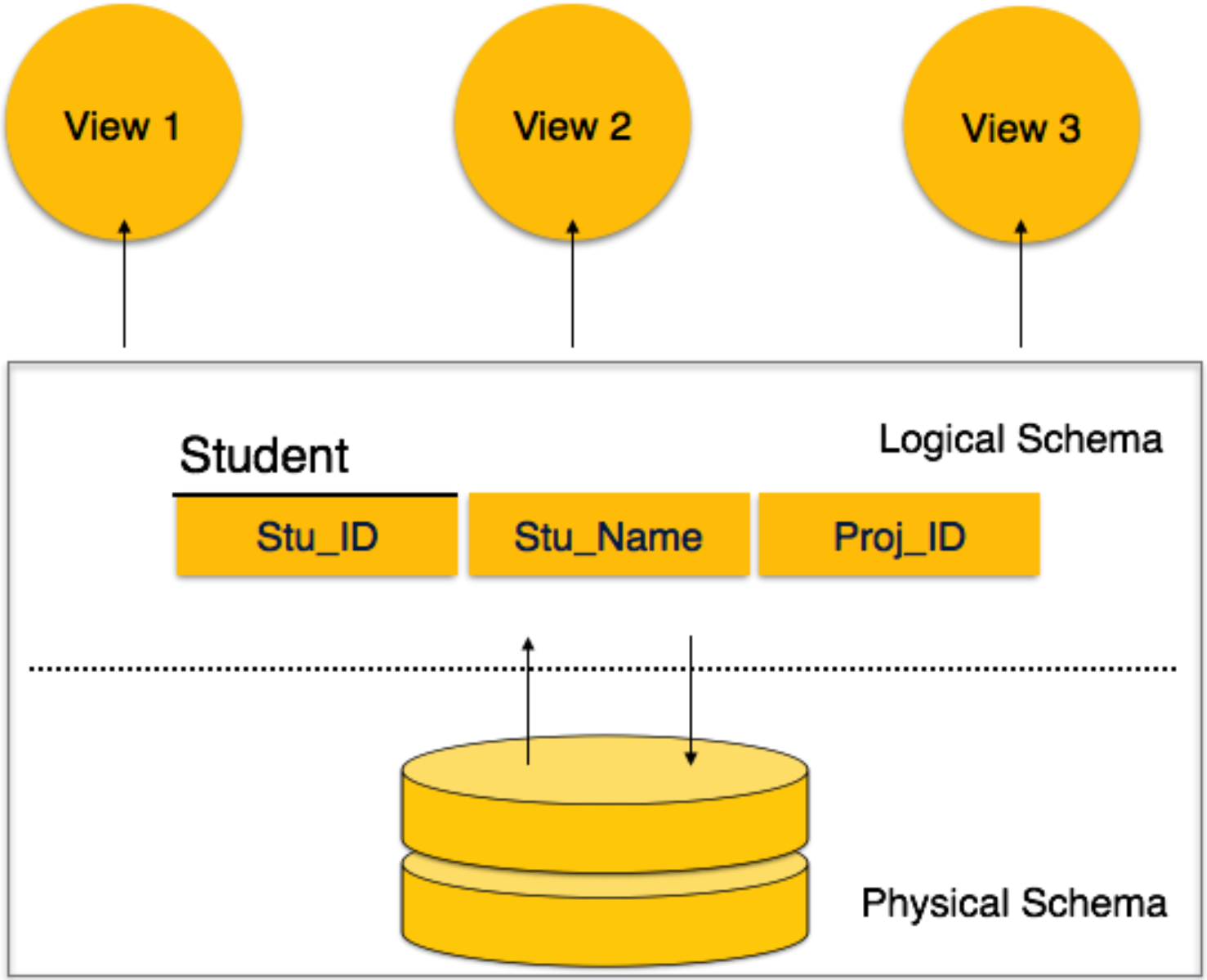


SID	SName	SAge	SClass	SSection
1101	Alex	14	9	A
1102	Maria	15	9	A
1103	Maya	14	10	B
1104	Bob	14	9	A
1105	Newton	15	10	B

Database Scheme

- A database scheme is the structure that represents the logical view of the entire database. It defines how the data is organized and how the relations among them are associated. It formulates all the constraints that are to be applied on the data.

- A database scheme defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams. It’s the database designers who design the schema to help programmers understand the database and make it useful.


View 1
View 2
View 3
Student
Logical Schema
Stu_ID
Stu_Name
Proj_ID
Physical Schema

- **Physical Database Scheme** − This scheme related to the actual storage of data and its form of storage like files, indices, etc.

  It defines how the data will be stored in a secondary storage.

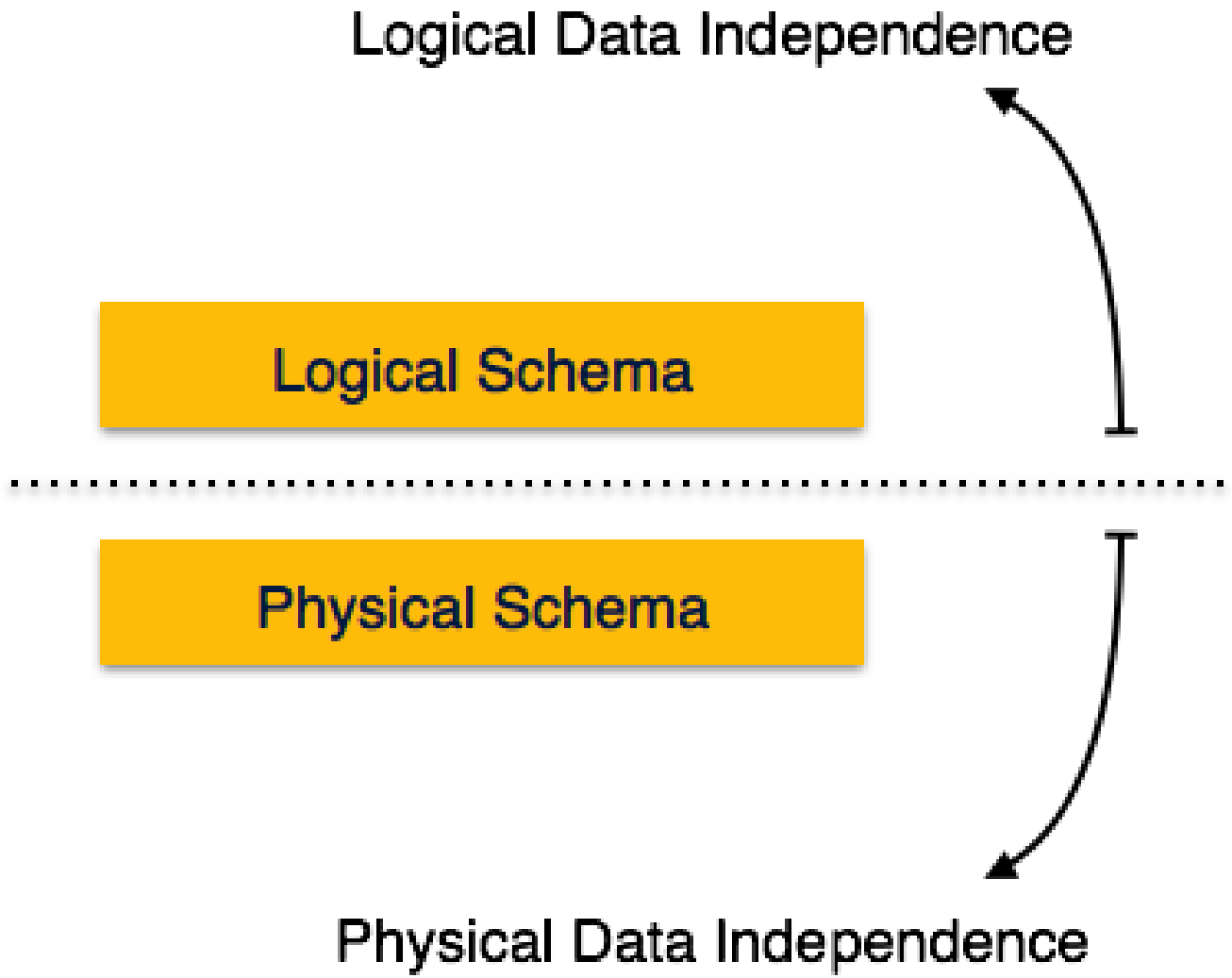
- **Logical Database Scheme** − This scheme defines all the logical part that need to be applied on the data stored. It defines tables, views, and integrity .

Data Independence

- A database system normally have a lot of data in addition to user's data.

For example, it stores data about data, to locate and bring back data easily. It is rather difficult to modify or update once it is stored in the database.

But as a DBMS expands, it needs to change over time to satisfy the requirements of the users. If the entire data is dependent, it would become a slow and highly complex job.

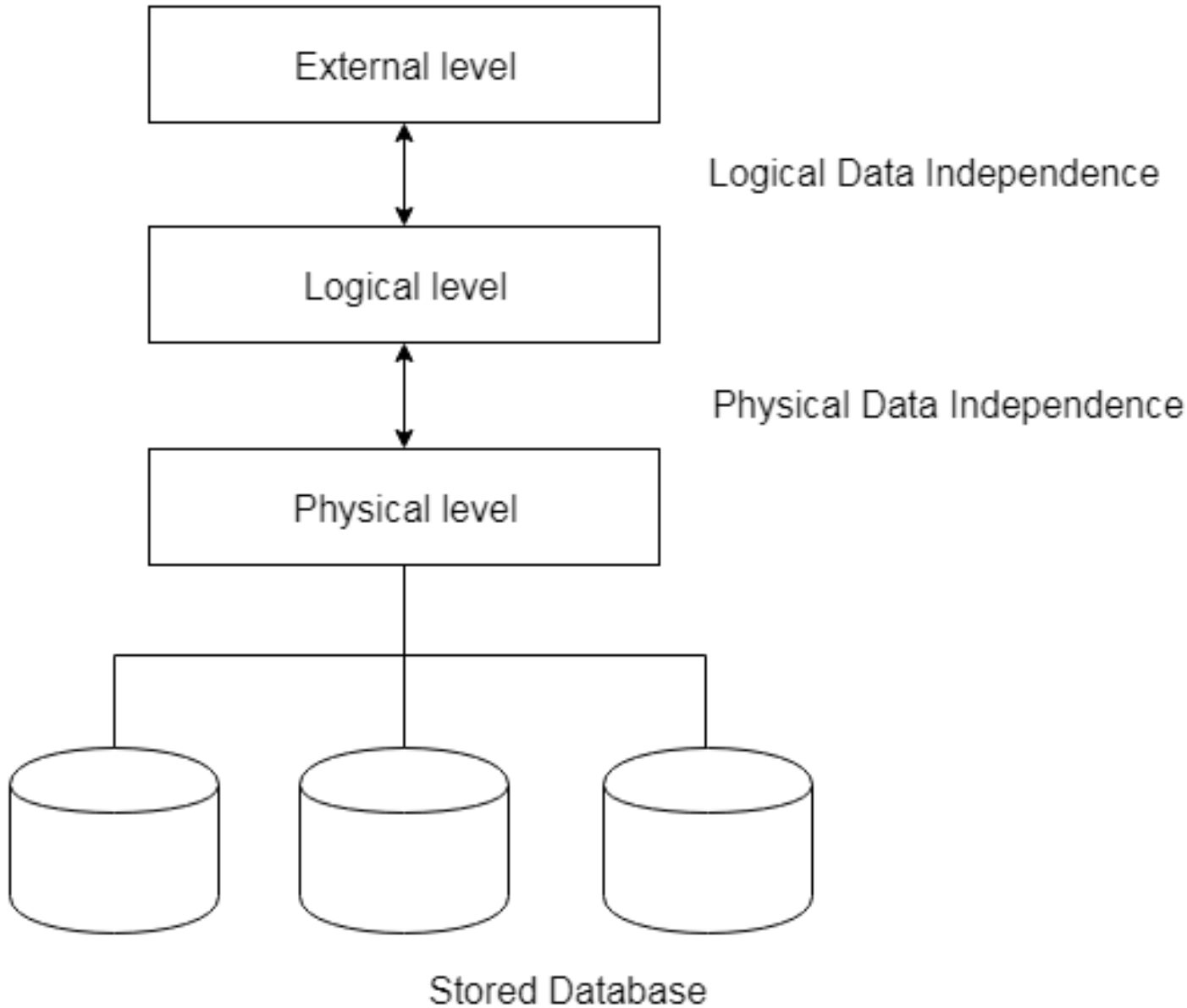

Logical Data Independence
Logical Schema
Physical Schema
Physical Data Independence

Logical Data Independence

- Logical data is data about database, that is, it stores information about how data is managed inside. For example, a table (relation) stored in the database and applied on that relation.
- Logical data independence is a kind of mechanism, which remove itself from actual data stored on the disk. If we do some changes on table format, it should not change the data on the disk.

Physical Data Independence

- All the scheme are logical and the actual data is stored in bit format on the disk. Physical data independence is the power to change the physical data without impacting the scheme or logical data.


External level
Logical Data Independence
Logical level
Physical Data Independence
Physical level
Stored Database

Attributes

- Entities are represented by means of their properties, called **attributes**. All attributes have values.

For example, a student entity may have name, class, and age as attributes.

- There exists a domain or range of values that can be assigned to attributes.

For example, a student's name cannot be a numeric value. It has to be alphabetic. A student's age cannot be negative, etc.

Types of Attributes

- **Simple attribute** – Simple attributes are atomic values, which cannot be divided further.

For example, a student's phone number is an atomic value of 10 digits.

- **Composite attribute** – Composite attributes are made of more than one simple attribute.

For example, a student's complete name may have first_name and last_name.

- **Derived attribute** – Derived attributes are the attributes that not in the physical database, but their values are derived from other attributes present in the database
- For example, age can be derived from data_of_birth.

- **Single-value attribute** − Single-value attributes contain single value. For example − pin or Security_Number.

- **Multi-value attribute** − Multi-value attributes may contain more than one values.

For example, a person can have more than one phone number, email_address, etc.

Entity-Set and Keys

Key is an attribute or collection of attributes that uniquely identifies an entity among entity set.

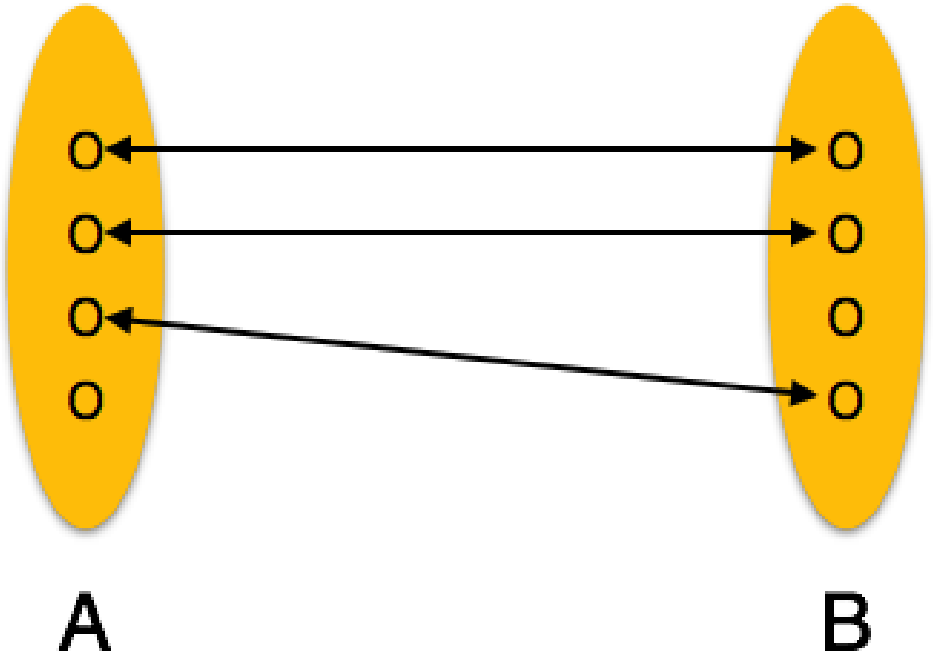
For example, the roll_number of a student makes him/her identifiable among students.

- **Super Key** − A set of attributes (one or more) that collectively identifies an entity .
- **Candidate Key** − A minimal super key is called a candidate key. An entity set may have more than one candidate key.
- **Primary Key** − A primary key is one of the candidate keys chosen by the database designer to uniquely identify the entity set.

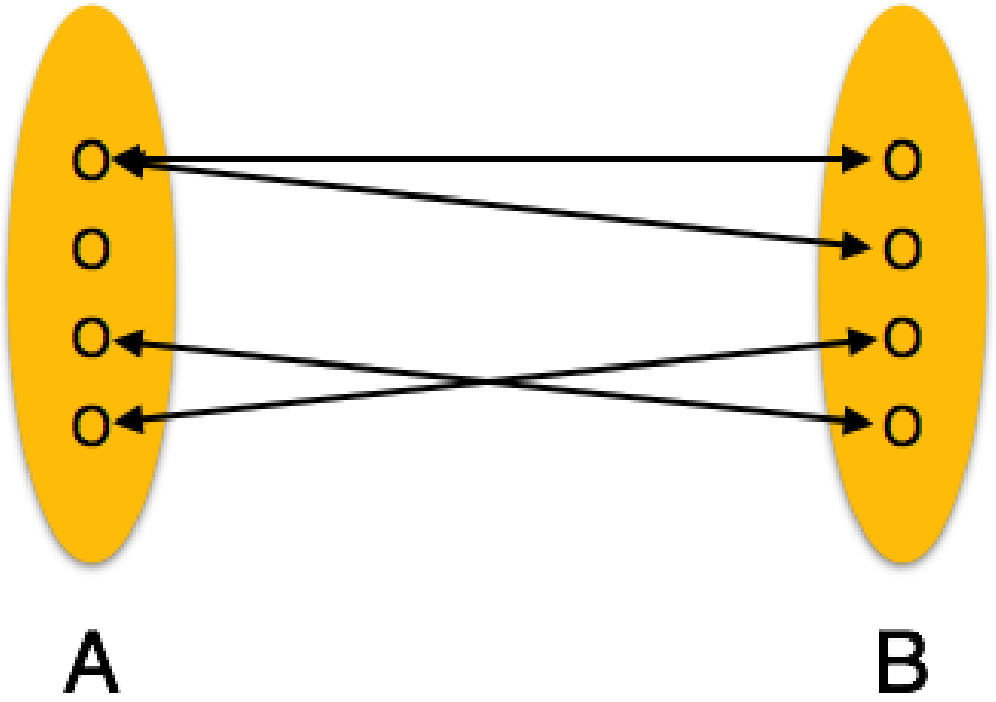
Mapping Cardinalities

- **Cardinality** defines the number of entities in one entity set, which can be associated with the number of entities of other set via relationship set.

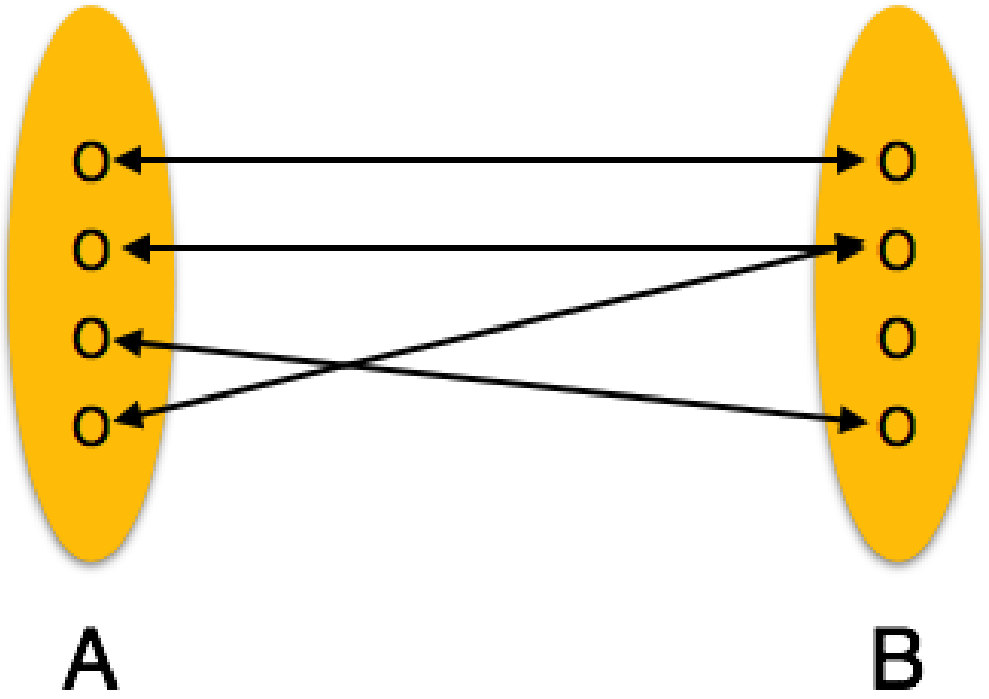
- **One-to-one** – One entity from entity set A can be associated with at most one entity of entity set B and vice versa.



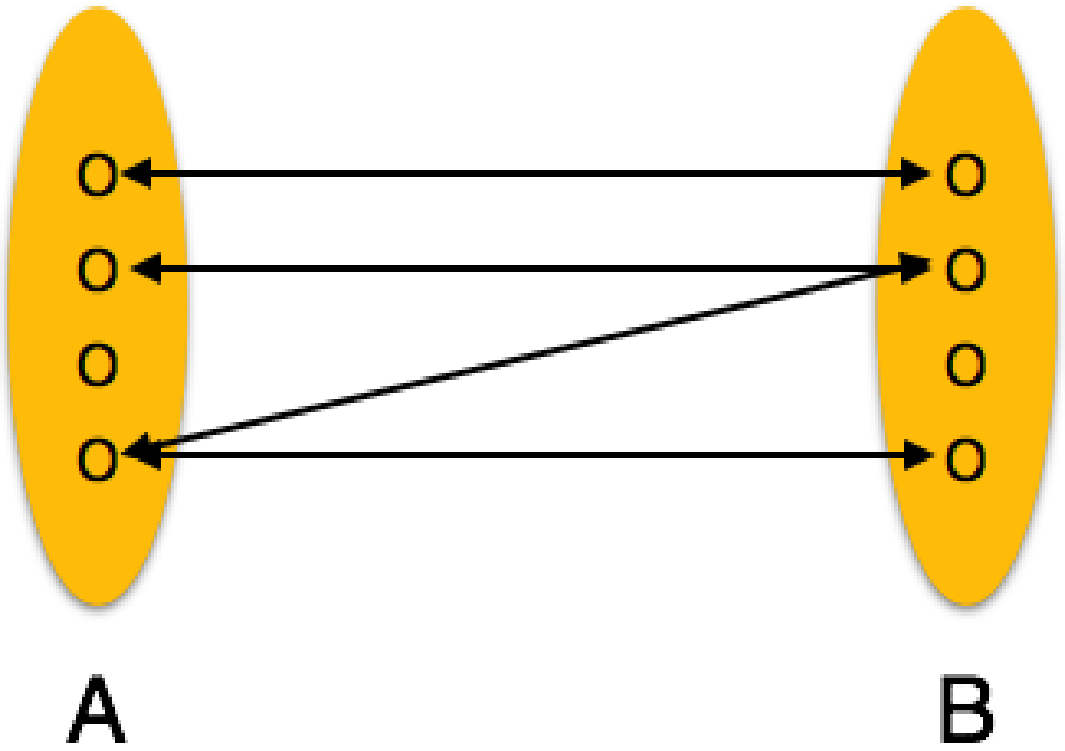
- **One-to-many** – One entity from entity set A can be associated with more than one entities of entity set B however an entity from entity set B, can be associated with at most one entity.



- **Many-to-one** − More than one entities from entity set A can be associated with at most one entity of entity set B, however an entity from entity set B can be associated with more than one entity from entity set A.



- **Many-to-many** – One entity from A can be associated with more than one entity from B and vice versa.



Advantages of DBMS

- **Controls database need:** It can control data because it stores all the data in one single database file and that recorded data is placed in the database.
- **Data sharing:** In DBMS, the authorized users of an organization can share the data among multiple users.
- **Easily Maintenance:** It can be easily maintainable due to the centralized nature of the database system.
- **Reduce time:** It reduces development time and maintenance need.
- **Backup:** It provides backup and recovery subsystems which create automatic backup of data from hardware and software failures and restores the data if required.
- **multiple user interface:** It provides different types of user interfaces like graphical user interfaces, application program interfaces

Disadvantages of DBMS

- **Cost of Hardware and Software:** It requires a high speed of data processor and large memory size to run DBMS software.
- **Size:** It occupies a large space of disks and large memory to run them efficiently.
- **Complex:** Database system creates additional complex and requirements.
- **Higher impact of failure:** Failure is highly impacted the database because in most of the organization, all the data stored in a single database and if the database is damaged due to electric failure or database corruption then the data may be lost forever.

Database Security

Introduction

Database and functions can be managed by two different modes of security controls:

- Authentication
- Authorization

Authentication

- Authentication is the process of confirming that a user logs in only in the rights to perform the activities he is authorized to perform. User authentication can be performed at operating system level or database level itself. By using authentication tools for biometrics such as retina and figure prints are in use to keep the database from hackers or malicious users.
- The database security can be managed from outside the database system. Here are some type of security authentication process:
- Based on Operating System authentications.
- Lightweight Directory Access Protocol (LDAP)
- For DB, the security service is a part of operating system as a separate product. For Authentication, it requires two different credentials, those are user id and password.

Authorization

- You can access the Database and its functionality within the database system, which is managed by the Database manager.
- Authorization is a process managed by the Database manager. The manager obtains information about the current authenticated user, that indicates which database operation the user can perform or access.

Here are different ways of permissions available for authorization:

- **Primary permission**: Grants the authorization ID directly.
- **Secondary permission**: Grants to the groups and roles if the user is a member
- **Public permission**: Grants to all users publicly.
- **Context-sensitive permission**: Grants to the trusted context role.

Authorization can be given to users based on the categories below:

- System-level authorization
- System administrator [SYSADM]
- System Control [SYSCTRL]
- System maintenance [SYSMAINT]
- System monitor [SYSMON]

System administration authority (SYSADM)

It is highest level administrative authority at the instance-level. Users with SYSADM authority can execute some databases and database manager commands within the instance.

Users with SYSADM authority can perform the following operations:

- Upgrade a Database
- Restore a Database
- Update Database manager configuration file.

System control authority (SYSCTRL)

It is the highest level in System control authority. It provides to perform maintenance and operations against the database manager process.

These operations can affect system resources, but they do not allow direct access to data in the database.

Users with SYSCTRL authority can perform the following actions:

- Updating the database, Node, or Distributed Connect Service (DCS) directory
- Forcing users off the system-level
- Creating or Dropping a database-level
- Creating, altering, or dropping a table space
- Using any table space
- Restoring Database

System maintenance authority (SYSMAINT)

It is a second level of system control authority. It provides to perform maintenance and operations against the database manager process. These operations affect the system resources without allowing direct access to data in the database. This authority is designed for users to maintain databases within a database manager instance that contains sensitive data.

Only Users with SYSMAINT or higher level system authorities can perform the following tasks:

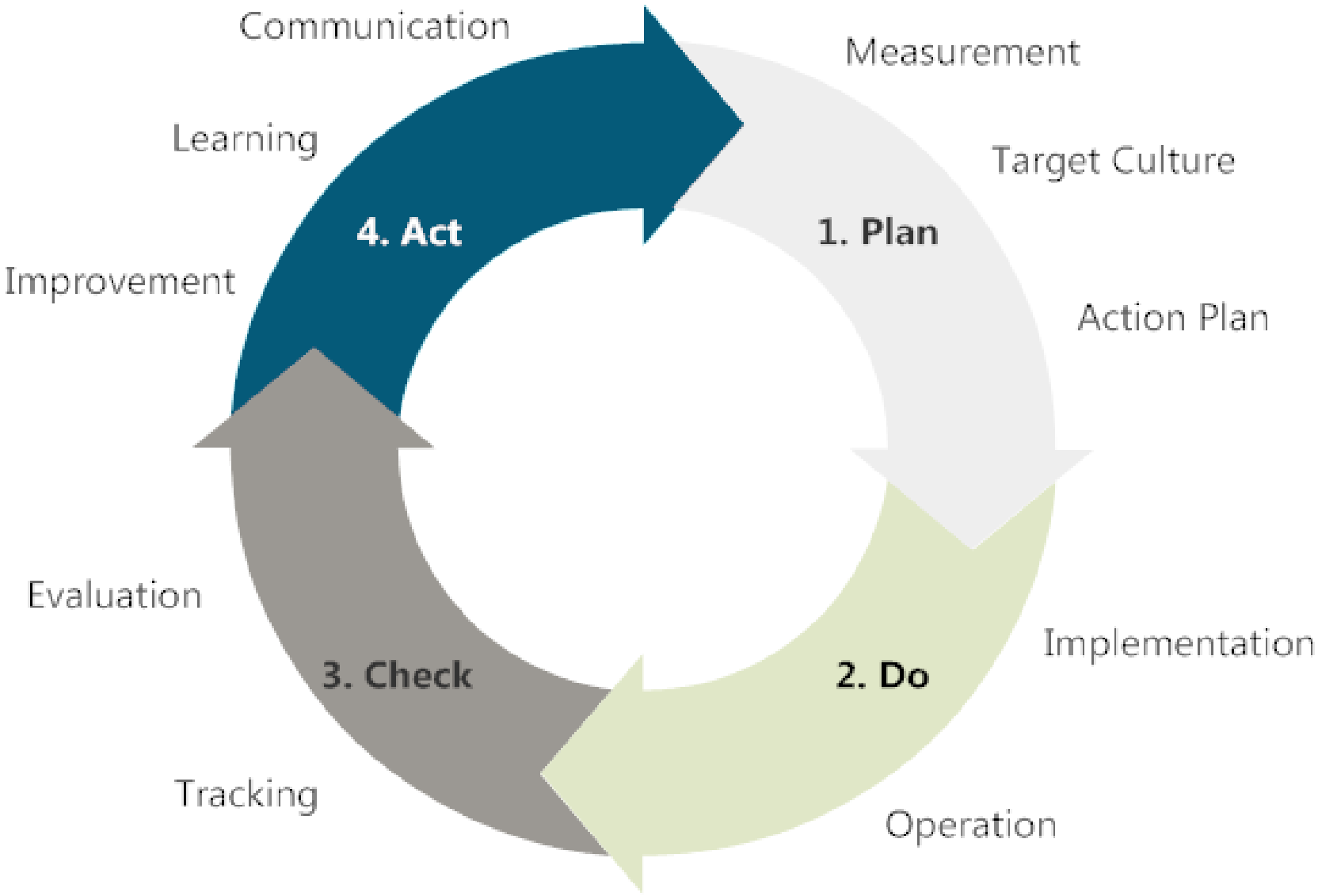
- Taking backup
- Restoring the backup
- Roll forward recovery
- Starting or stopping instance
- Restoring table spaces
- Executing db command
- Taking system monitor snapshots in case of an Instance level user or a database level user.

System monitor authority (SYSMON)

With this authority, the user can monitor or take snapshots of database manager process. SYSMON authority enables the user to run the following tasks:

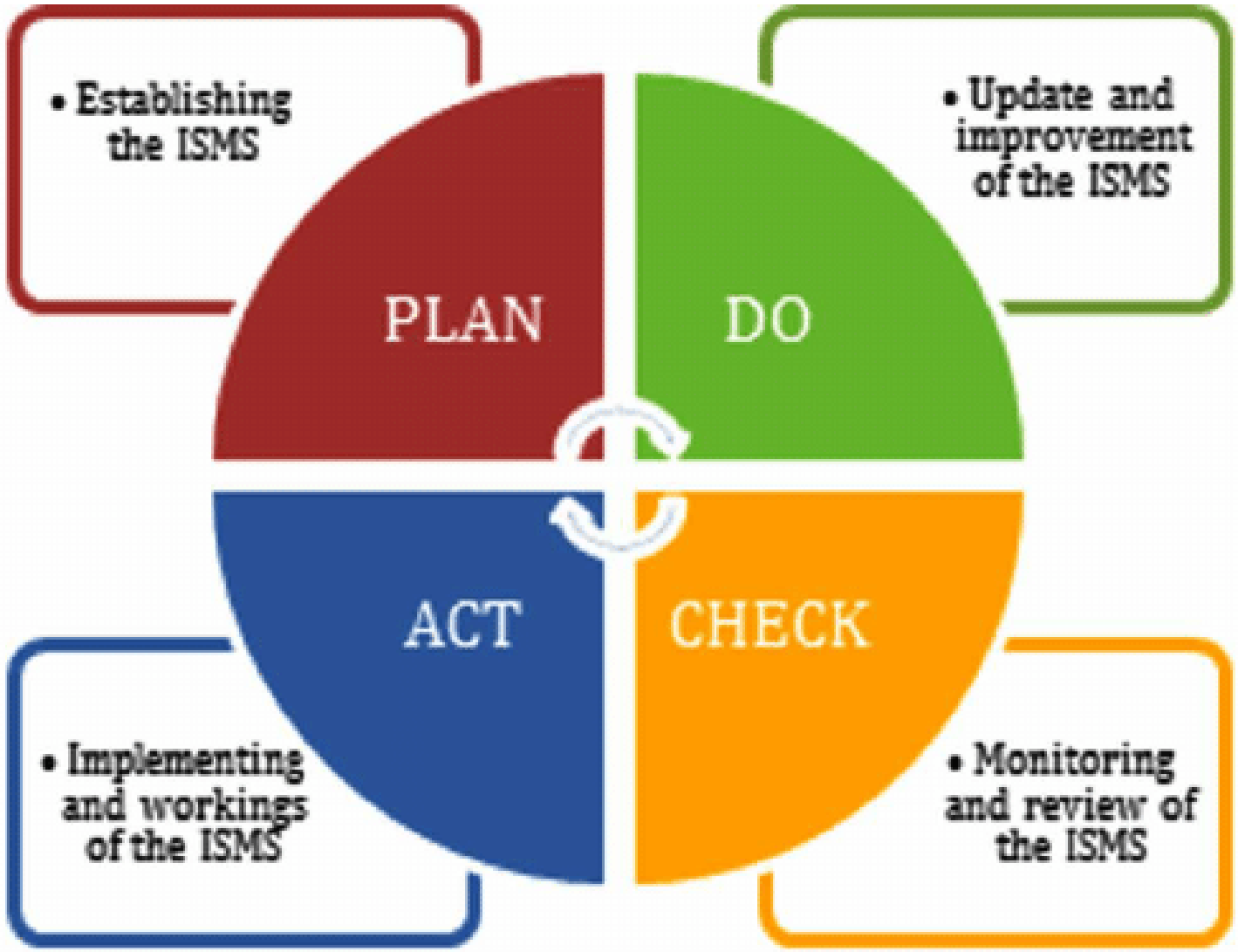
- GET DATABASE MANAGER MONITOR SWITCHES
- GET MONITOR SWITCHES
- GET SNAPSHOT
- LIST
  - LIST ACTIVE DATABASES
  - LIST APPLICATIONS
  - LIST DATABASE PARTITION GROUPS
  - LIST DCS APPLICATIONS
  - LIST PACKAGES
  - LIST TABLES
  - LIST TABLESPACE CONTAINERS
  - LIST TABLESPACES
  - LIST UTITLITIES
- RESET MONITOR
- UPDATE MONITOR SWITCHES


IT Security
Management
Security
Ctrl


Communication
Measurement
Learning
Target Culture
4. Act
1. Plan
Improvement
Action Plan
Evaluation
Implementation
3. Check
2. Do
Tracking
Operation

International standard for an Information Security Management System

PLAN	Establish ISMS policy, objectives, processes in the organisation To manage risk and improve security.
DO	Implement and operate the ISMS policy, controls and processes
CHECK	Assess and measure process performance against policy and objectives
ACT	Take corrective and preventative action based on the results of ISMS audit, management review or other relevant information to enable continuous improvement.


• Establishing the ISMS
PLAN
DO
• Update and improvement of the ISMS
ACT
CHECK
• Implementing and workings of the ISMS
• Monitoring and review of the ISMS

Establishing the ISMS

PLAN

Define scope and boundaries of the ISMS
- depends on the organisation

Define ISMS Policy

Define the risk assessment approach
- there are different methods for risk assessment

Identify the risks

Evaluate the risks

Identify and evaluate options for the treatment of risks

Select control objectives and controls for the treatment of risks

Obtain management approval for the proposed risks

Obtain management to implement and operate the ISMS

Prepare a statement

Implement and operate the ISMS policy, controls and processes

A risk treatment plan that identifies management action, resources, responsibilities and managing the IS risks

DO

Implement the risk treatment plan

Implement controls

Implement training and awareness programs

Manage operation of the ISMS

Manage resources

Implement to detect and respond to security incidents

Assess and measure process performance against policy and objectives (Monitor and review)

Execute monitoring process

Undertake regular review of ISMS effectiveness

CHECK

Measure the effectiveness of controls to verify security requirements

Review risk assessments at planned intervals with residual risks
(taking into account changes in organisation, technology, business objectives, processes, identified threats etc)

Conduct internal ISMS audits at planned intervals

Undertake regular management reviews of ISMS

Record actions and events that could impact the effectiveness / performance of ISMS

Maintain and Improve the ISMS

ACT

Implement identified improvements

Take corrective / preventative action and apply lessons learned (including other organisations)

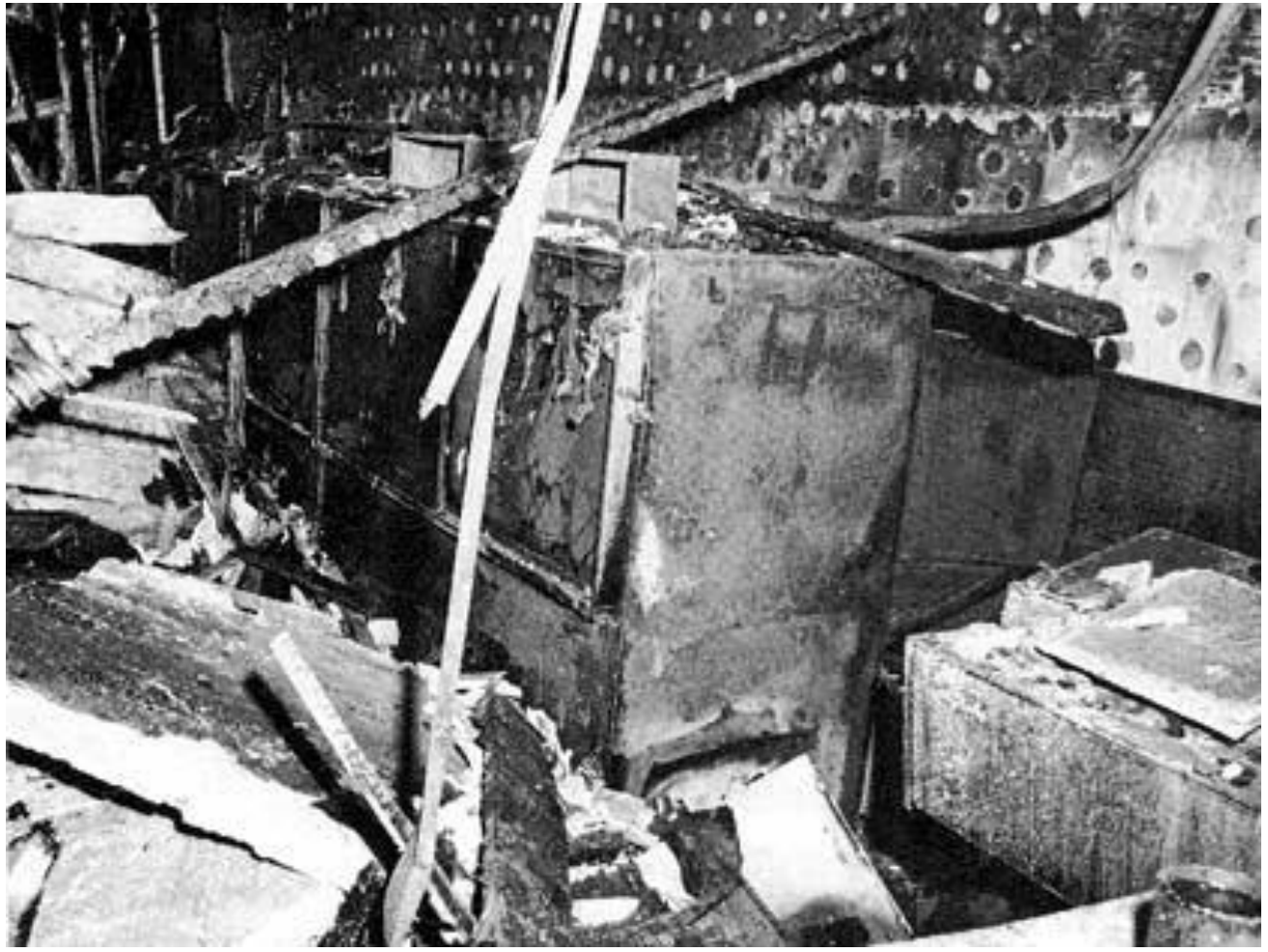
Communicate the actions to interested parties

Ensure improvements achieve their intended objectives

Natural threats to system components

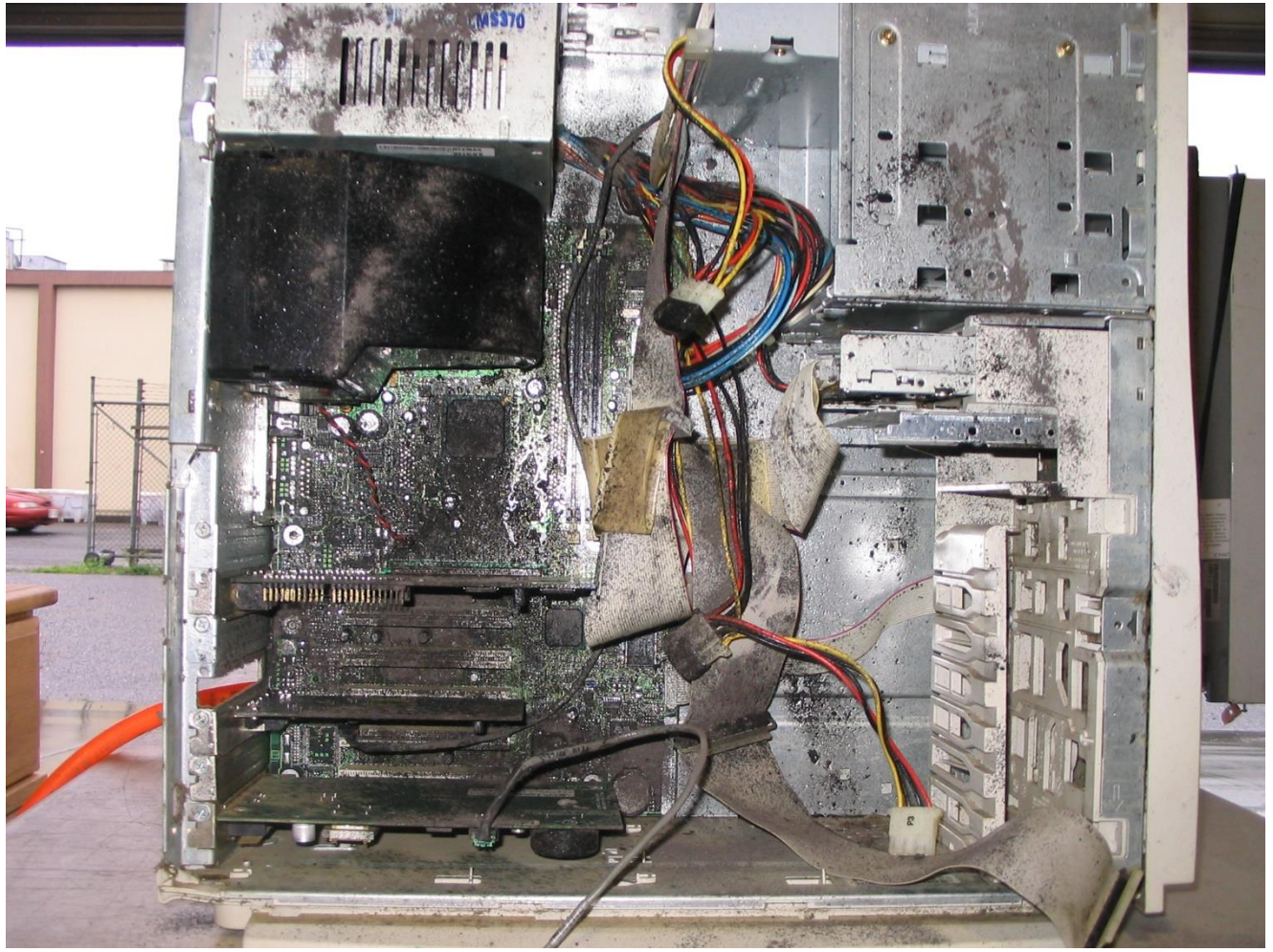
- Fire
- Acts of Nature / Flood
- Smoke
- Explosion and Impact
- Loss of essential services

FIRE



FLOOD

SMOKE Damage

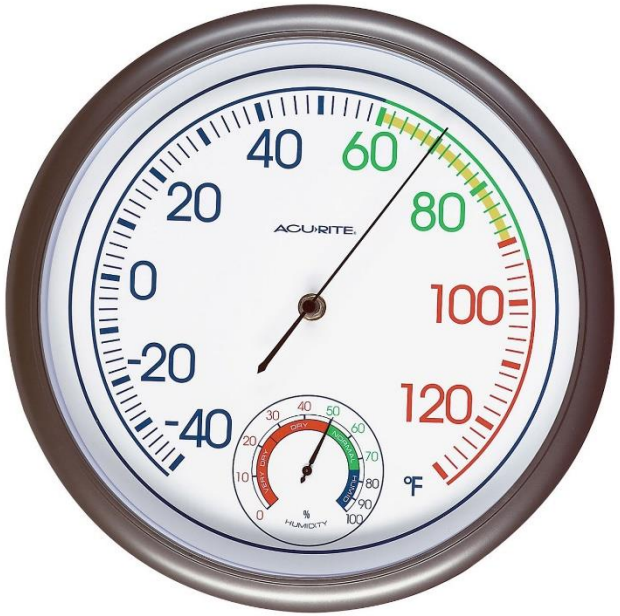


Explosion or Impact

Neither of these events are kind to computers !

Loss of essential services

Computers need :



The correct temperature **The correct humidity** **and electricity**

Air and water supplies

to Work

Human threat to system components

Terrorist attack

Malicious action

Errors

Loss of key staff

Industrial action

The greatest threat to IT

Physical threats through nature of location

Goals of Physical Security

Protect people (staff and visitors) from harm on site

Protect IT Assets from theft or damage

Protect against unauthorised access to equipment,
IT installations, Electronic media and documentation

Physical threat identification

What are the risks to the organisation?

External Physical Threats

Internal Physical Threats

Human Physical Threats

Choice of location

IT Security not often considered in choice of premises
Older established companies may have started in normal area,
But are located in premises

Site Selection

Given the opportunity to consult in new premises then the ideal would be:

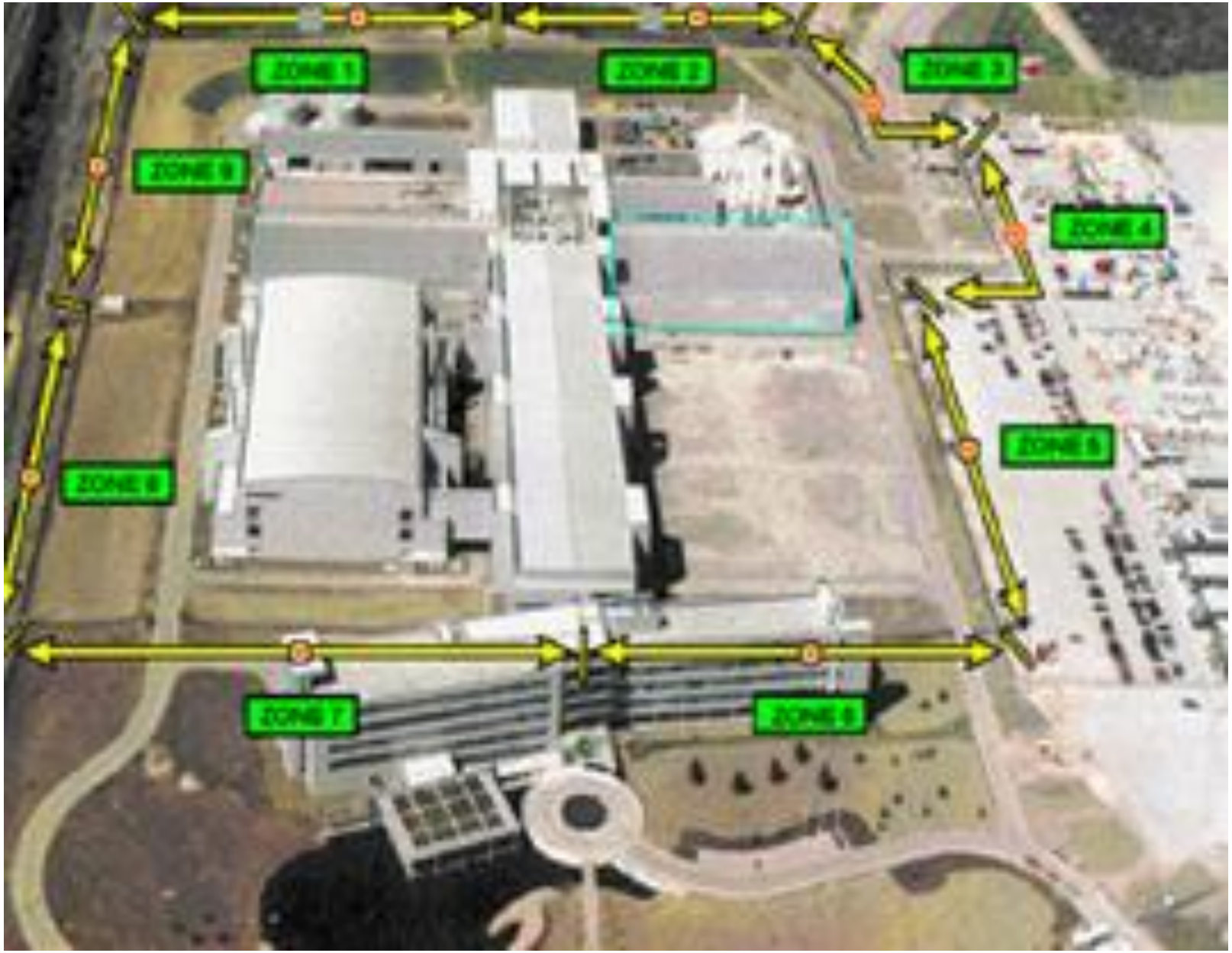
A low natural disaster area
Easy access to emergency services – fire, police, hospitals
Buildings with strong protection
Hidden external cabling and services
Security with controlled access

Planning Physical Security Measures

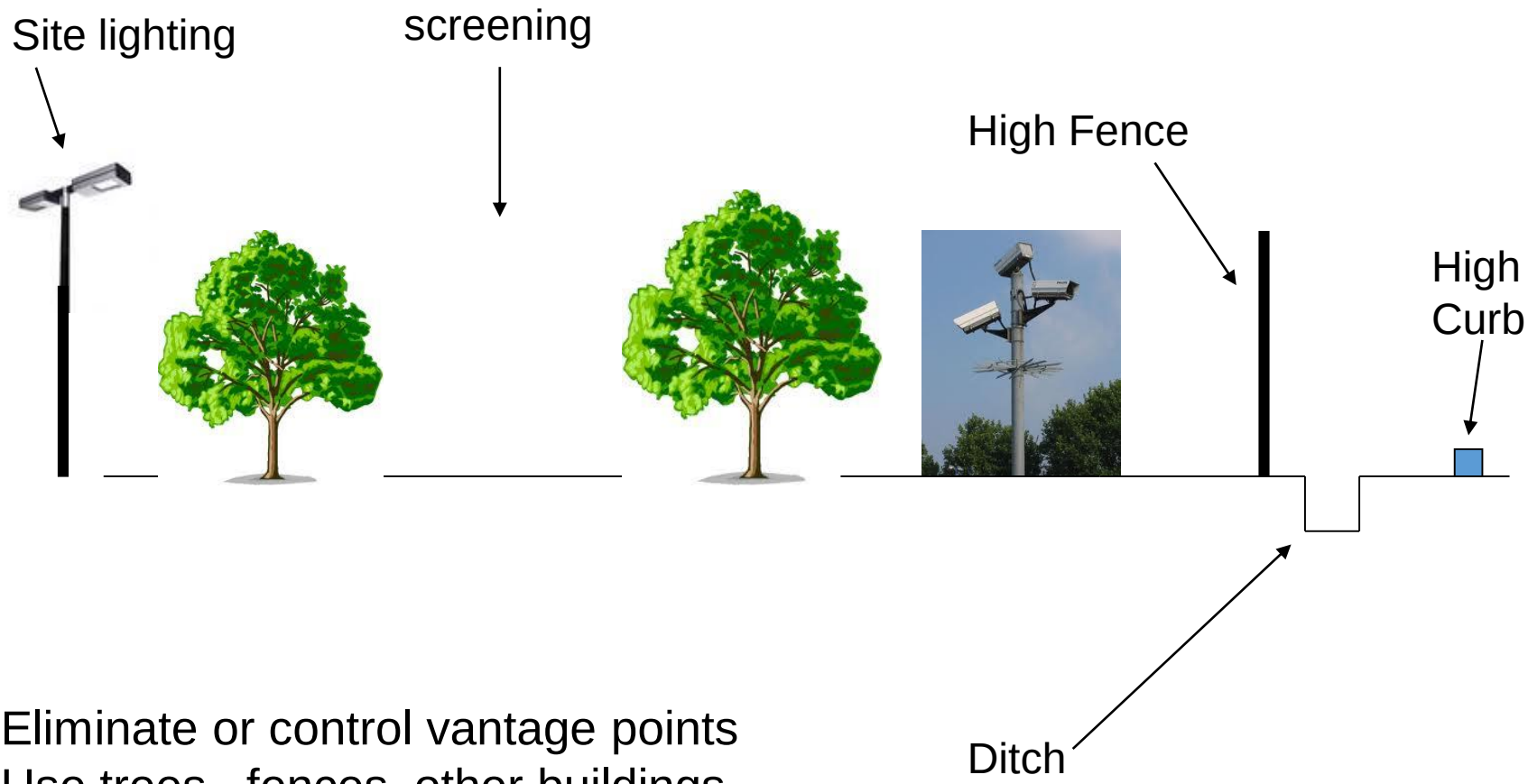
• **Response** – procedures and actions for responding to a breach.

• **Recovery** – your plan to continue business and operations as normally as possible following an incident. Mitigation planning is part of your response and recovery with the aim of minimizing the effects of any incident.

• **Re-assessment** – Before implementing any changes, you need to revisit your strategic plan to ensure that goals and objectives will be met.


ZONE 1
ZONE 2
ZONE 3
ZONE 4
ZONE 5
ZONE 6
ZONE 7
ZONE 8

Perimeter + Building Protection



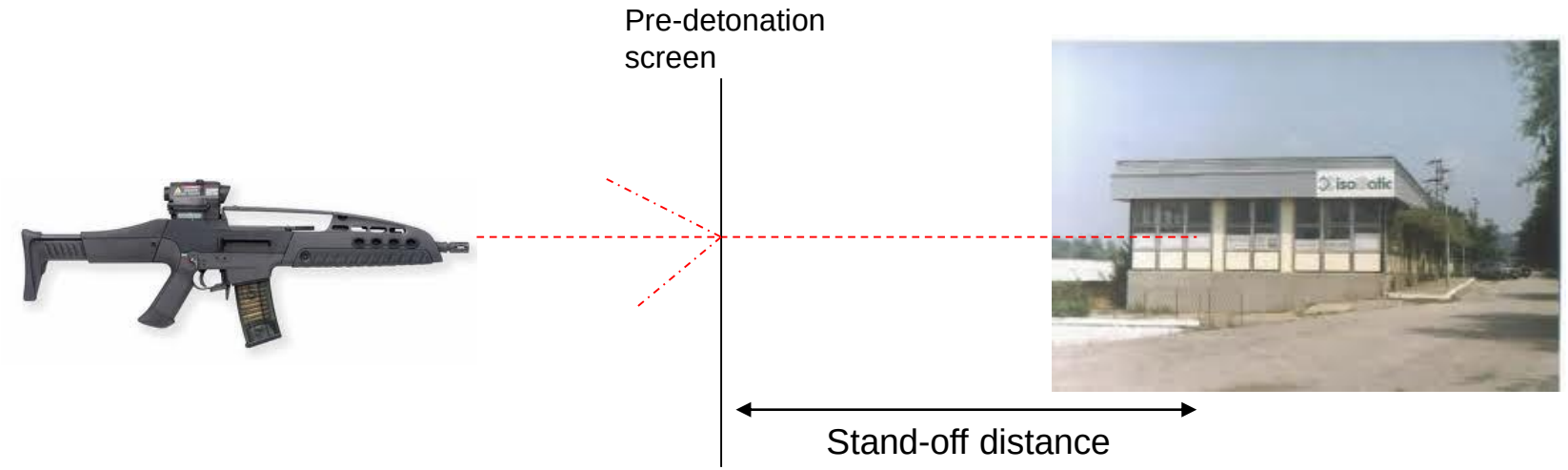
Eliminate or control vantage points
Use trees, fences, other buildings

Lighting should deter, but prevent glare that may blind guards.

Guards +
Patrol Dogs

Perimeter + Building Protection

Protection from stand-off weapons



Obstruct line-of-sight from vantage points outside site
Use pre-detonation screen

Perimeter + Building Protection

Basic Building Security

Locks.
Self-closing doors.
Window locks.
Lockable internal doors between rooms.
Alarms.

More advanced

Complex locks
One way screws
Device sensors (light or pressure)

Building Protection

Fire detection

Heat sensors
Smoke Detectors
Auto-dial fire alarms

Wet pipe
Dry pipe
sprinkler

Building Protection

Electric Power

Needs:

- Emergency power-off controls
- Voltage maintenance
- Surge protection
- Back-up power

Also humidity control:

- less than 40% increases
- static = damage potential

Uninteruptable power supply

Building Protection

Un-interruptible Power Supply

Range of systems
Dependent on needs

Internal Physical Security

Protect the server

Keep servers in a locked room and control access to it.

Individual servers should have their own lock and key to stop people opening them.

Networking switch boards should be protected by a lock to control which ports are activated.

Unplug unused extensions.

Internal Physical Security

Protect information on paper

Consider lockable filing cabinets.

Save personal information before throwing it out.

Have a ‘clear-desk’ policy so that employees lock up sensitive papers when they are not working on them.

Encourage users to pick up their documents from printers, faxes and photocopiers machine .

Internal Physical Security

Monitor visitors

Don’t let visitors inside the secure perimeter without an escort.

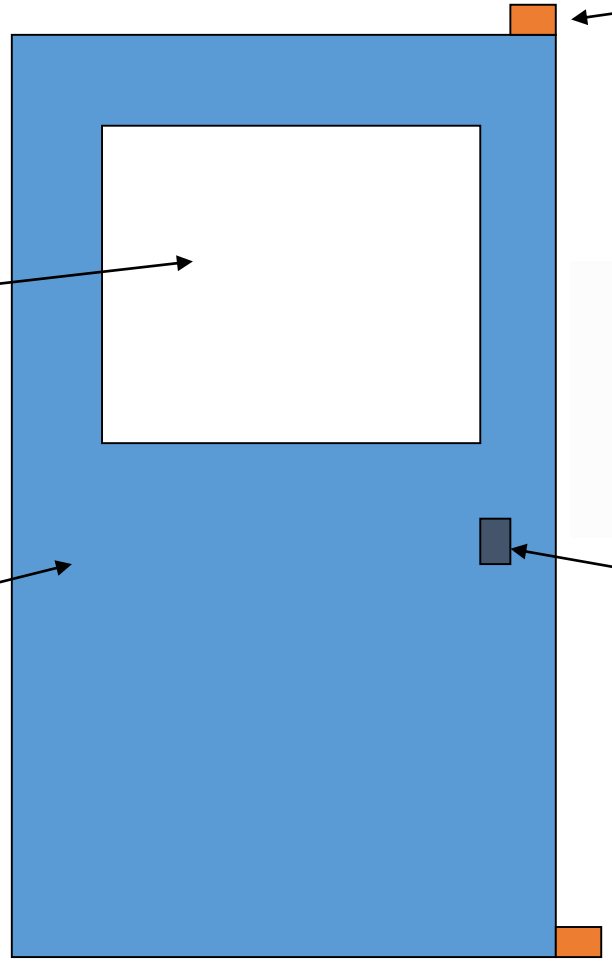
Vet contractors and support personnel.

Restrict access to sensitive areas, such as server rooms or HR records.

Staff should be encouraged to query strangers in secure areas.

Internal Physical Security

Secure Entry to Sensitive areas



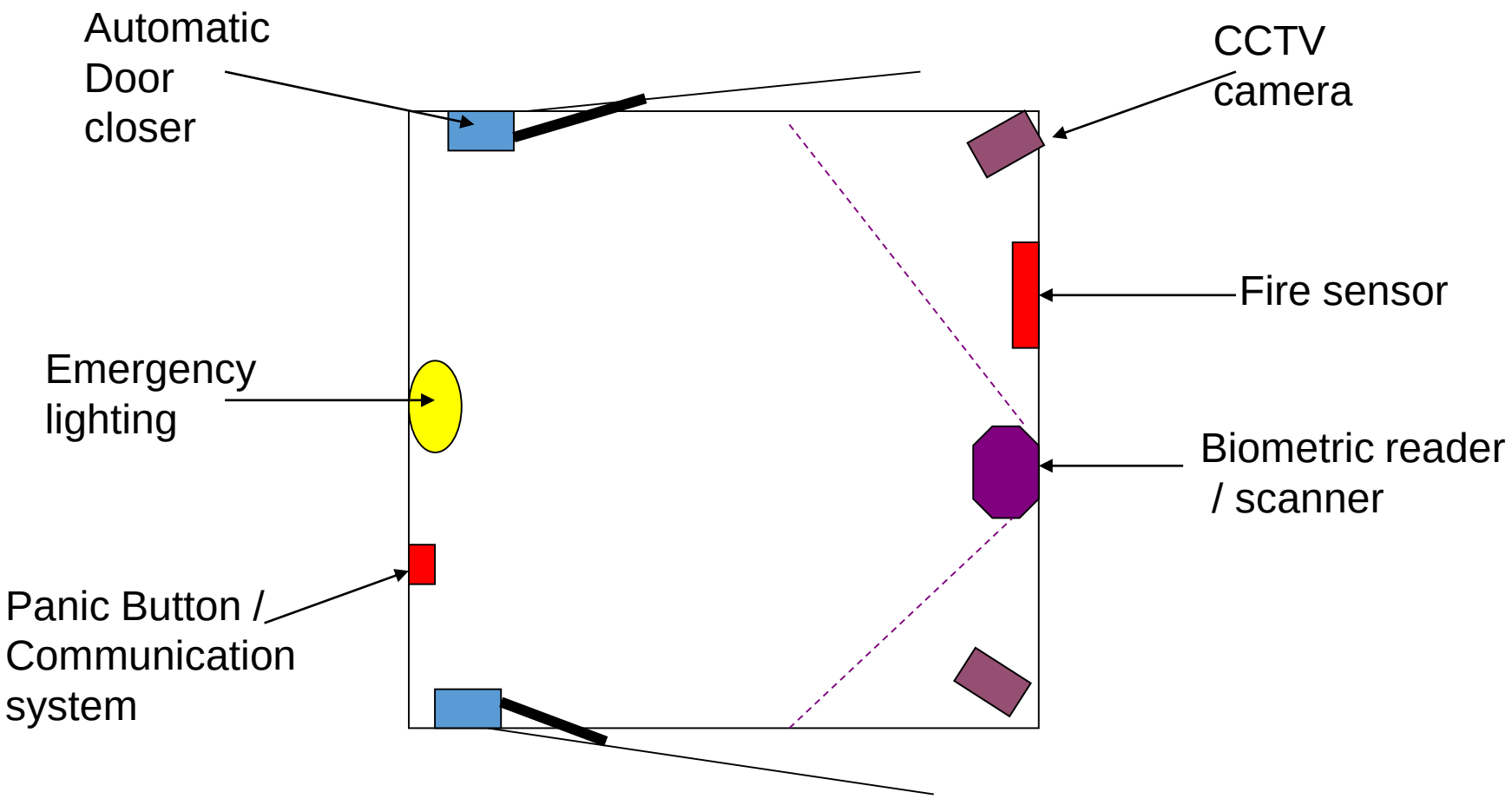
Door closed sensor

Key card reader
+ optional keypad

Bullet proof window

Fireproof / bullet proof
Metal door + frame

Mechanical over-ride lock



Secure entry chamber

Sensitive Compartmented Information Facility (SCIF)

SCIF Door with Hardware and Alarm Panels

SCIF Specification

Vault classification

	Floors	Walls	Ceiling
A Top Secret	8” RC	8”RC	8”RC
B Secret	4” RC	8”RC	4”RC
C Confidential	4” C	8”RC	4”RC

Min compression strength 3000psi after 28 days ageing
5/8” diameter steel reinforcing bars laid 6” on centres

Preferably windowless else must be secured with steel bars embedded into the masonry.

Entrance GSA class 6 Vault door

Walls clad with soundproof material

Internal IDS
- structural vibration sensors
- Point sensors
- passive ultrasonic sensors
-- volumetric motion sensors

External IDS
- Fence sensors
- Laser sensors (light beams)

Alarms
perimeter door
Vault door
internal space

Vents and ducts >6”
Windows <18ft from ground level
Tamper switches

SCIF specifications that may be considered for wider implementation

LOCKS

Locks + Keys (high security spec)
Programmable locks (mechanical or electronic combination)
- Change codes frequently !

ID Cards

Photo ID
Wireless proximity
Magnetic strip
Smart cards using pin number with card
Biometric devices

Movement control

2 person rule

General office controls

Entry controls
Office layout
Property controls
Electronic media control
Clear desk policy

Physical security information management (PSIM)

Is a category of software that provides a platform and applications designed to integrate multiple unconnected security applications and devices and control them through one comprehensive user interface.

It collects and correlates events from existing disparate security devices and information systems (video, access control, sensors, analytics, networks, building systems, etc.) to empower personnel to identify and proactively resolve situations.

Very Secure Data Centres