Mobile Databases

Why Mobile Databases?

- Number of smartphones in use around the world passed 2.9 billion in 2018 (Ref. Statista)

- Next billion devices could be reached within less than two years.

- More businesses move toward employees mobility.

- Powerful lightweight computing devices and low cost mobile connectivity paved the way for data-driven applications.

Why Mobile Databases?

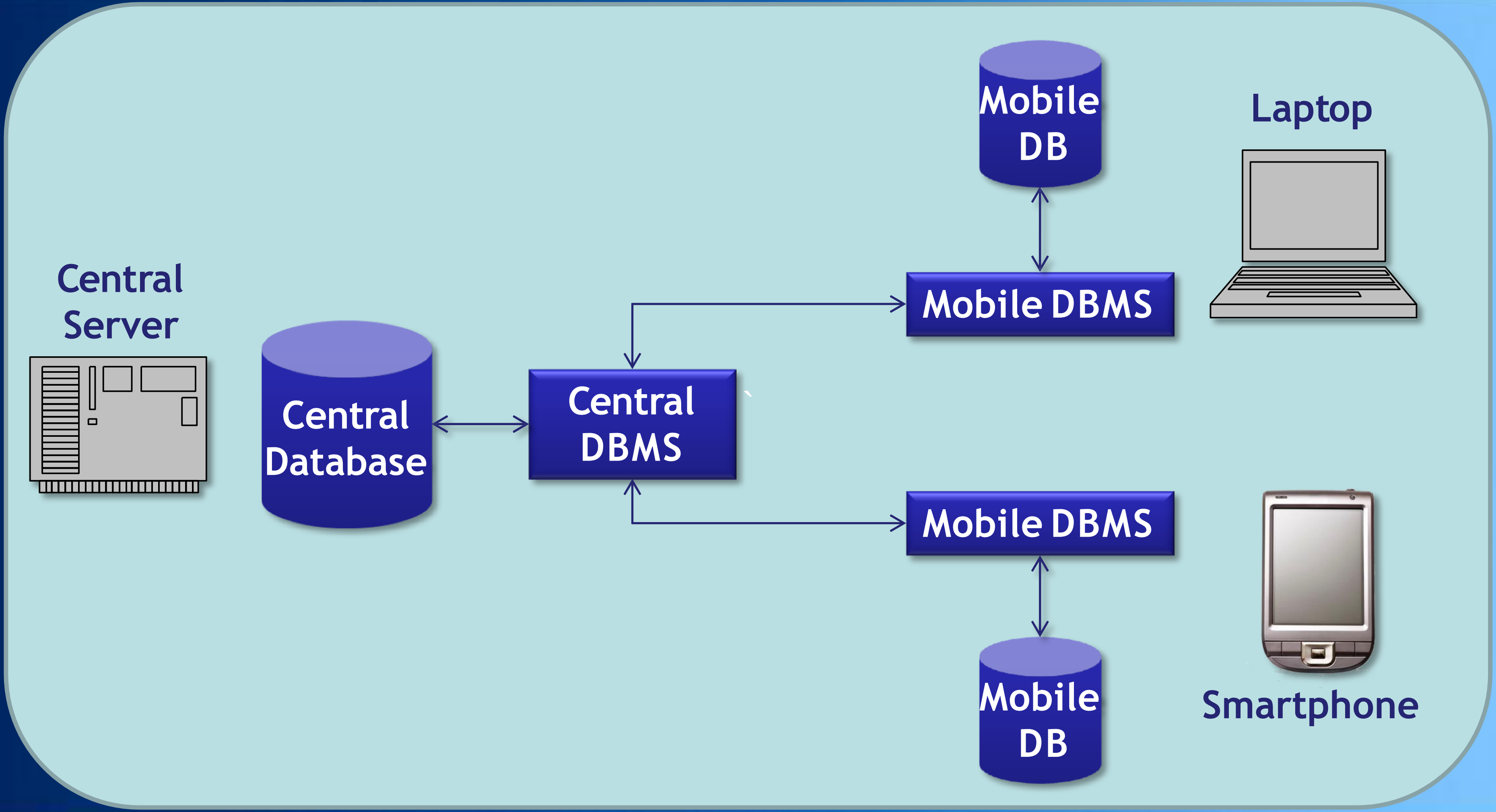
- ➢ Mobile data-driven applications enable us to access any data from anywhere, anytime.
- ➢ Examples:
  - ✓ Salespersons can update sales records on the move.
  - ✓ Reporters can update news database anytime.
  - ✓ Doctors can retrieve patient’s medical history from anywhere.
- ➢ Mobile DBMSs are needed to support these applications data processing capabilities.

Mobile Database:

- A **mobile database** is a database that can be connected to by a mobile computing device over a wireless mobile network.

- Mobile databases:
  - ✓ Physically separate from the central database server.
  - ✓ Resided on mobile devices.
  - ✓ Capable of communicating with a central database server or other mobile clients from remote sites.
  - ✓ Handle local queries without connectivity.

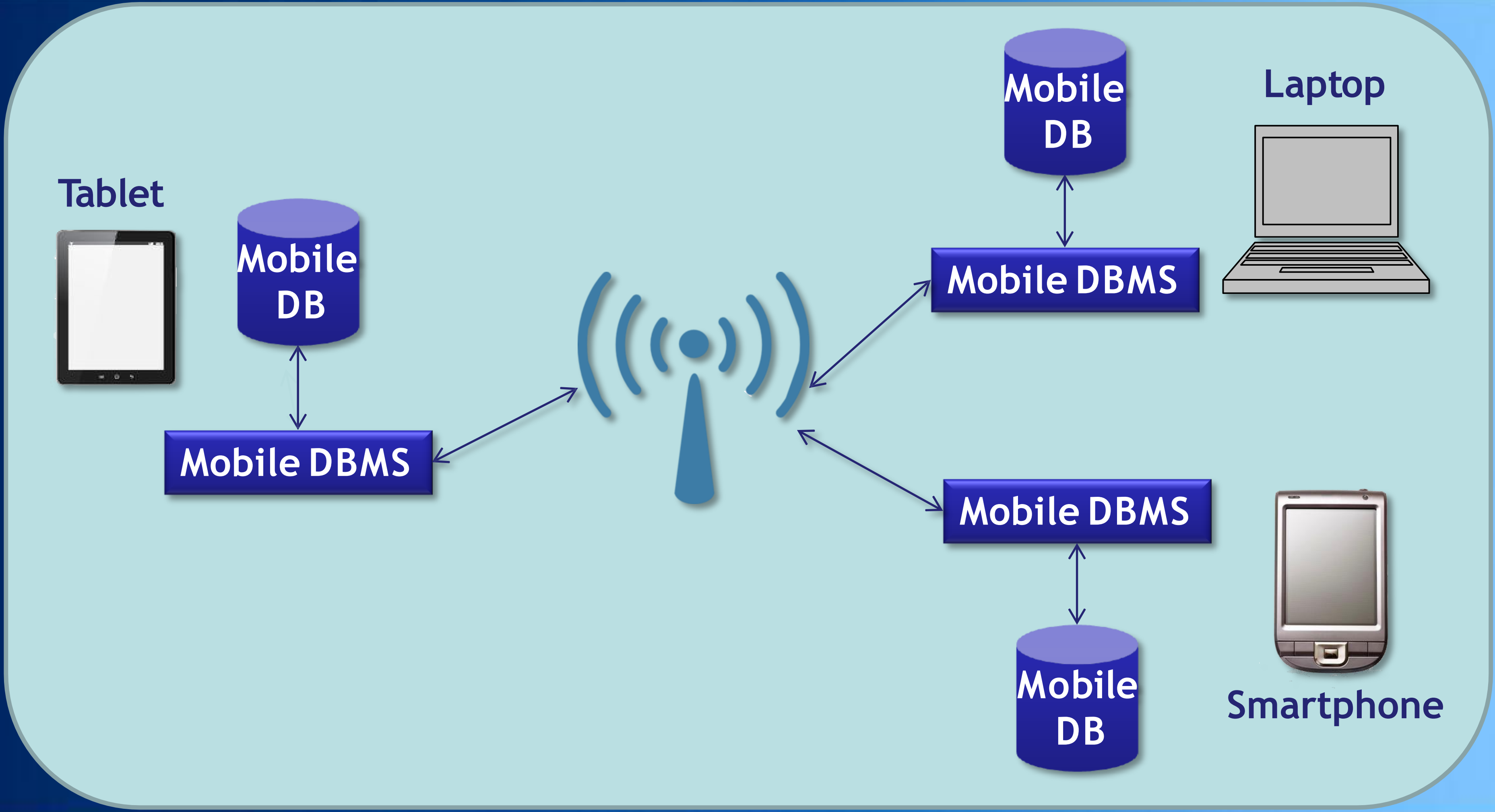
Client-Server Mobile Databases:



Client-Server Mobile Databases:

- Client-server model is the traditional model of information systems.
- It is the dominant model for existing mobile databases.
- The server can become a single point of failure and performance bottleneck.
- Even storing data on a cluster of machines to backup central database might cause performance bottleneck and data inconsistency.

Peer-to-Peer Mobile Databases:



Peer-to-Peer Mobile Databases:

- In P2P mobile databases, the database maintenance activities are distributed among clients.

- Every process plays part of the role of the server, besides its client role.

- A client that wants to access a piece of data, sends a request to other peer clients and they forward the request until the data is found.

- The major problem in this model is ensuring the availability of data .

Characteristics of Mobile Environments:

Why not use an appropriate existing model of databases in a mobile environment?

- Characteristics of mobile environments:
  - ✓ Restricted bandwidth of wireless networks.
  - ✓ Limited power supply.
  - ✓ Limited resources.
  - ✓ Mobility.
  - ✓ Disconnections.

Current Approach:

- Currently most mobile application developers use “flat files” to store application data.
- A “flat file” is a file containing records that have no structured interrelationship.
- Advantages:
  - ✓ Smaller and easier to manage.
- Disadvantages:
  - ✗ Applications need to know the organization of the records within the file.
  - ✗ Developers have to implement the required database functionalities.

Requirements of Mobile DBMSs:

- Mobile DBMSs should satisfy the following requirements :

  - ✓ Small memory footprint.
  - ✓ Flash-optimized storage system.
  - ✓ Data synchronization.
  - ✓ Security.
  - ✓ Low power consumption.
  - ✓ Self-management.
  - ✓ Embeddable in applications.

Small Memory Footprint:

- Memory footprint is amount of main memory that an application uses while running.

- Mobile devices have limited memory, so the mobile database application should have a small footprint.
- The size of mobile database affects the overall application footprint.
- Mobile DBMSs should be customizable to include only the required database functionalities.

Flash-Optimized Storage System:

- Flash memories are dominant storage devices for portable devices .
- They have feature such as:
  - ✓ Small size.
  - ✓ Better shock resistance.
  - ✓ Low power consumption.
  - ✓ Fast access time.
  - ✓ No mechanical seek and rotational latency.

- Mobile DBMSs need to be optimized to exploit the advantages of the new storage devices.

Data Synchronization:

- Portable devices cannot stay connected all the time.

- Users can access and manipulate data on their devices.

- They are also unable to store a large amount of data due to lack of storage capacity.

- Mobile DBMSs should have the synchronize functionality to integrate different versions of data into a consistent version.

Security:

- Security is very important for data-centric mobile applications.

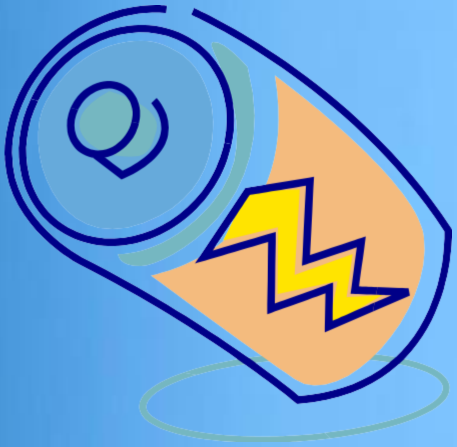
- It is more important when the application works with critical data that its disclosure results in potential loss or damage.

- Data that are transmitted over a wireless network are more prone to security issues.

- Mobile DBMSs should implement a complete end-to-end security to ensures the secure transfer of data.

Low Power Consumption:

- Portable devices have limited power supplies.



- Battery life of mobile phones is expected to increase only 20% over the next 10 years.
- Processor, display and network connectivity are the main power consumers in a mobile device.
- Mobile DBMSs need to be optimized for efficient power consumption.

Self-Management:

- In traditional databases, the database administrator (DBA) is responsible for databases maintenance.

- In mobile DBMSs there can be no DBA to manage the database.
- Mobile DBMSs need to support self-management and automatically perform the DBA tasks.
- Some mobile DBMSs allow remote management that enables a DBA to manage the mobile databases from a remote location.

Embeddable in applications :

- Administrators does not have direct access to mobile devices.

- Mobile DBMSs should be an integral part of the application that can be delivered as a part of the applications.

- The database must be embeddable as a Dynamic Link Library-DLL (windows) /Shared Object-SO (Linux) file in the applications.

- It must be also possible to deploy the database as a stand-alone DBMS with support of multiple transaction.

Existing Mobile Databases:

- **Mobile databases:**
  - ✓ Sybase SQL Anywhere
  - ✓ Oracle Lite
  - ✓ Microsoft SQL Server Compact
  - ✓ SQLite
  - ✓ IBM DB2 Everyplace (DB2e)
  - ✓ Realm DB
  - ✓ ORMLite
- **Embedded database:**
  - ✓ TinyDB
  - ✓ PicoDBMS

Sybase SQL Anywhere:

- Initially created by Watcom as Watcom SQL.
- SQL Anywhere was launched in in 1995.
- It dominates the mobile-database field, with about 68% of the mobile database market.
- Database files are independent of the operating system and transferable between supported platforms.
- Strong encryption is supported for both database files and client-server communication.

Oracle Lite:

- Omniscience Object Technology, Inc. was acquired by Oracle Corporation in November 1996 .

- Their product (Omniscience ORDBMS) became the first version of Oracle Lite.

- Oracle Lite runs in under 1 MB of memory, and can be installed in 3 MB of hard disk space.

- Personal Oracle Lite (POL) is a lightweight, single-user relational database that runs on desktops, laptops, down to the smallest hand help devices.

Microsoft SQL Server Compact:

- Formerly known as SQL Server Mobile Edition.
- SQL Server Compact is free to download and redistribute.

Microsoft® SQL Server®

- It is optimized for an architecture where all applications share the same memory pool.
- SQL CE runs in-process with the application which is hosting it.
- It has a memory footprint of approximately 5 MB and disk footprint of less than 2 MB .

SQLite:

- SQLite is an open source mobile database engine.
- It is a server-less database engine that needs zero-configuration.
- SQLite is a popular choice as mobile database for local storage in mobile applications.
- SQLite engine has no standalone processes with which the application program communicates.
- SQLite implements most of the SQL-92 standard.

IBM DB2 Everyplace:

- DB2e has been discontinued and April 2013 is the end of support date.
- It had the biggest market share after SQL Anywhere.
- It had the smallest memory footprint (350 KB) in compare to other commercial mobile databases.
- IBM has replaced DB2e with IBM solidDB family.
- SolidDB is a in-memory MDBMS with robust data catching features.

Embedded Databases:

- Embedded database systems are tightly integrated with an application that requires access to stored data.
- They are hidden from the application’s end-user and requires little or no ongoing maintenance.
- Embedded databases need less resources in compare with mobile databases.
- They are optimized for specific devices such as smartcards and sensors.
- They support limited and specified functionalities of the standard SQL.

PicoDBMS:

- PicoDBMS only supports sufficient functionalities for smartcard applications.
- Smartcard applications are used for data management such as insert, delete, update and search.
- PicoDBMS supports a part of SQL:
  - ✓ INSERT, UPDATE, DELETE, SELECT
  - ✓ CREATE/DROP TABLE/VIEW
  - ✓ GRANT/REVOKE
- Footprint size of PicoDBMS is about 30KBytes.

TinyDB:

- TinyDB has been developed at University of Berkeley.
- It supports only essential functionalities for sensor applications.
- Most of the sensor applications are used to filter out some data so they just need to select data with given conditions.
- TinyDB supports only SELECT operation of the standard SQL.
- Its memory footprint is only 3KBytes.

Comparison:

Target Devices		Mobile DBMSs
Extremely Small Devices with Low Computing Power	Sensors	TinyDB
	Smartcards	PicoDBMS
Small Devices with High Computing Power	Cell Phones, PDAs, Car Navigators, Ultra Books	Sybase SQL Anywhere, Oracle Lite, MS SQL Server CE, SQLite IBM DB2 Everyplace

Functionalities:

	TinyDB	PicoDBMS	Oracle Lite	IBM DB2e	MS SQL Server Compact
Minimum Footprint Size	3 KB	30 KB	970 KB	320 KB	2 MB
SQL	SELECT only	a part of SQL99	a part of SQL99	a part of SQL99	a part of SQL99
Views	N	Y	Y	Y	Y
Integrity Constraints	N	N/A	Y	Y	Y
Concurrency	N	N	Y	Y	Y
Indexing	N	Y	Y	Y	Y
Encryption	N	N/A	Y	Y	Y
Access Control	N	Y	Y	Y	Y

Supportability of MDBMS Requirements:

	TinyDB	PicoDBMS	Oracle Lite	IBM DB2e	MS SQL Server Compact
Small Footprint	Y	Y	Y	Y	Y
Flash-Optimized Storage System	N	N	N	N	N
Data Synchronization	N	N	Y	Y	Y
Self-Management	Y	Y	N/A	Y	N/A
Low Power Consumption	Y	Y	N	Y	Y
Security	N	Y	Y	Y	Y

References:

1. Nori, A. Mobile and embedded databases. In Proc. *SIGMOD 2007*, ACM Press (2007), 1175-1177.

2. Tsiftes, N. and Dunkels, A. A database in every sensor. In Proc. *SenSys 2011*, ACM Press (2011), 316-332.

3. Whang, K.Y., Song, I.Y., Kim, T.Y., and Lee, K.H. The ubiquitous DBMS. *ACM SIGMOD Record 38*, 4 (2009), 14-22.

4. Sen, R. DBMS techniques for lightweight computing devices. In Proc. *MobiDE 2011*, ACM Press (2011), 1-8.

Thank You