

## WHAT IS OLTP:-

**Online transaction processing**, or **OLTP**, refers to a class of systems that facilitate and manage transaction-oriented applications, typically for data entry and retrieval transaction processing. The term is somewhat ambiguous; some understand a "transaction" in the context of computer or database transactions, while others (such as the Transaction Processing Performance Council) define it in terms of business or commercial transactions.<sup>[1]</sup> OLTP has also been used to refer to processing in which the system responds immediately to user requests. An automatic teller machine (ATM) for a bank is an example of a commercial transaction processing application.

The technology is used in a number of industries, including banking, airlines, mailorder, supermarkets, and manufacturing. Applications include electronic banking, order processing, employee time clock systems, e-commerce, and eTrading. The most widely used OLTP system is probably IBM's CICS.

In [computer science](#), **transaction processing** is information processing that is divided into individual, indivisible operations, called *transactions*. Each transaction must succeed or fail as a complete unit; it cannot remain in an intermediate state.

*definition –*

Databases must often allow the real-time processing of SQL transactions to support e-commerce and other time-critical applications. This type of processing is known as online transaction processing (OLTP).

OLTP (online transaction processing) is a class of program that facilitates and manages transaction-oriented applications, typically for data entry and retrieval transactions in a number of industries, including banking, airlines, mailorder, supermarkets, and manufacturers. Probably the most widely installed OLTP product is IBM's CICS (Customer Information Control System).

Today's online transaction processing increasingly requires support for transactions that span a network and may include more than one company. For this reason, new OLTP software uses client/server processing and brokering software that allows transactions to run on different computer platforms in a network.

A system consisting of interconnected computers that share a central storage system and various peripheral devices such as a printer, scanner, or router. Each computer connected to the system can operate independently, but has the ability to communicate with other external devices and computers.

### **REQUIREMENTS:-**

Online transaction processing increasingly requires support for transactions that span a network and may include more than one company. For this reason, new OLTP software uses client/server processing and brokering software that allows transactions to run on different computer platforms in a network.

In large applications, efficient OLTP may depend on sophisticated transaction management software (such as CICS) and/or database optimization tactics to facilitate the processing of large numbers of concurrent updates to an OLTP-oriented database.

For even more demanding Decentralized database systems, OLTP brokering programs can distribute transaction processing among multiple computers on a network. OLTP is often integrated into service-oriented architecture (SOA) and Web services.

### **BENEFITS:-**

Online Transaction Processing has two key benefits: simplicity and efficiency. Reduced paper trails and the faster, more accurate forecasts for revenues and expenses are both examples of how OLTP makes things simpler for businesses.

### **ADVANTAGES OF OLTP:-**

Online Transaction Processing (OLTP) has the following advantages:

- It provides faster and more accurate forecast for revenues and expenses.
- It provides a concrete foundation for a stable organization because of timely modification of all transactions.
- It makes the transactions much easier on behalf of the customers by allowing them to make the payments according to their choice.
- It broadens the customer base for an organization by simplifying and speeding up individual processes.

### **DISADVANTAGES OF OLTP:-**

As with any information processing system, security and reliability are considerations. Online transaction systems are generally more susceptible to direct attack and abuse than their offline counterparts. When organizations choose to rely on OLTP, operations can be severely impacted if the transaction system or database is unavailable due to data corruption, systems failure, or network availability issues. Additionally, like many modern online information technology solutions, some systems require offline maintenance which further affects the cost-benefit analysis.

Online Transaction Processing (OLTP) has the following disadvantages:

- It makes the database much more susceptible to intruders and hackers because it makes the database available worldwide.
- For B2B (business-to-business) transactions, businesses must go offline to complete certain steps of an individual process, causing buyers and suppliers to miss out on some of the efficiency benefits that the system provides. As simple as OLTP is, the simplest disruption in the system has the potential to cause a great deal of problems, causing a waste of both time and money.
- It can lead to server failure, which may cause delays or even wipe out large amounts of data from the database.

## **IMPLEMENTATION OF OLTP:-**

On-Line Transaction Processing is a processing that supports the daily business operations. Also known as operational processing and OLTP. An OLTP is a database which must typically allow the real-time processing of SQL transactions to support traditional retail processes, e-commerce and other time-critical applications. It is also a class of program that helps to manage or facilitate transaction oriented applications such as data entry and retrieval transactions in a number of industries, including banking, airlines, mail order, supermarkets, and manufacturers. With today's business environment, it is impossible to run a business without having to rely on data. Processing online transactions these days increasingly requires support for transactions spanning a large network or even the global internet and may include many companies.

1. Today, with the ubiquity of the internet, more and more people even from those remote areas are not doing transactions online through an e-commerce environment. The term transaction processing is often associated with the process wherein an online shop or e-commerce website accepts and processes payments through a customer's credit or debit card in real time in return for purchased goods and services.
2. During the process of online transactions, a merchant payment system will automatically connect to the bank or credit card company of the customer and carry out security, fraud and other checking for validity after which authorization to take the payment follows. It is strongly advised that when a company looks for other companies that will handle online transactions and processing, the company should have a system infrastructure that is robust, secure and reliable that give customers fast, seamless and secure checkout time.
3. An OLTP implementation tends to be very large involving very high volume of data at any given time. Business organizations have invested in sophisticated transaction management software like Customer Information Control System (CICS) and database optimization tactics that can help OLTP process very large

numbers and volumes of concurrent updates on an OLTP-oriented database.

4. There are also many OLTP brokering programs which can distribute transaction processing among multiple computers on a network that can enhance the functions of an OLTP working on a more demanding decentralized database system. Service oriented architectures and web services are now commonly integrated with OLTP.
5. The two main benefits with using OLTP are simplicity and efficiency. OLTP helps simplify a business operation by reducing paper trails and helping draw faster and more accurate forecasting for revenues and expenses. OLTP helps provide a concrete foundation with timely updating of corporate data. For an enterprise' customers, OLTP allows the more choices on how they want to pay giving them more flexible time and enticing them to make more transactions. Most OLTP transactions offer services to customers 24 hours a day seven days a week .

### **APPLICATION OF OLTP:-**

OLTP applications are often used to capture new data or update existing data. An order-entry system is a typical example of an OLTP application.

OLTP applications have the following characteristics:

- Transactions that involve small amounts of data
- Indexed access to data
- Many users
- Frequent queries and updates
- Fast response times.

### **Compare Data Warehouse database and OLTP database?**

The **data warehouse** and the **OLTP** data base are both relational databases. However, the objectives of both these databases are different.

- The **OLTP database** records transactions in real time and aims to automate clerical data entry processes of a business entity.

Addition, modification and deletion of data in the OLTP database is essential and the semantics of the application used in the front end impact on the organization of the data in the database.

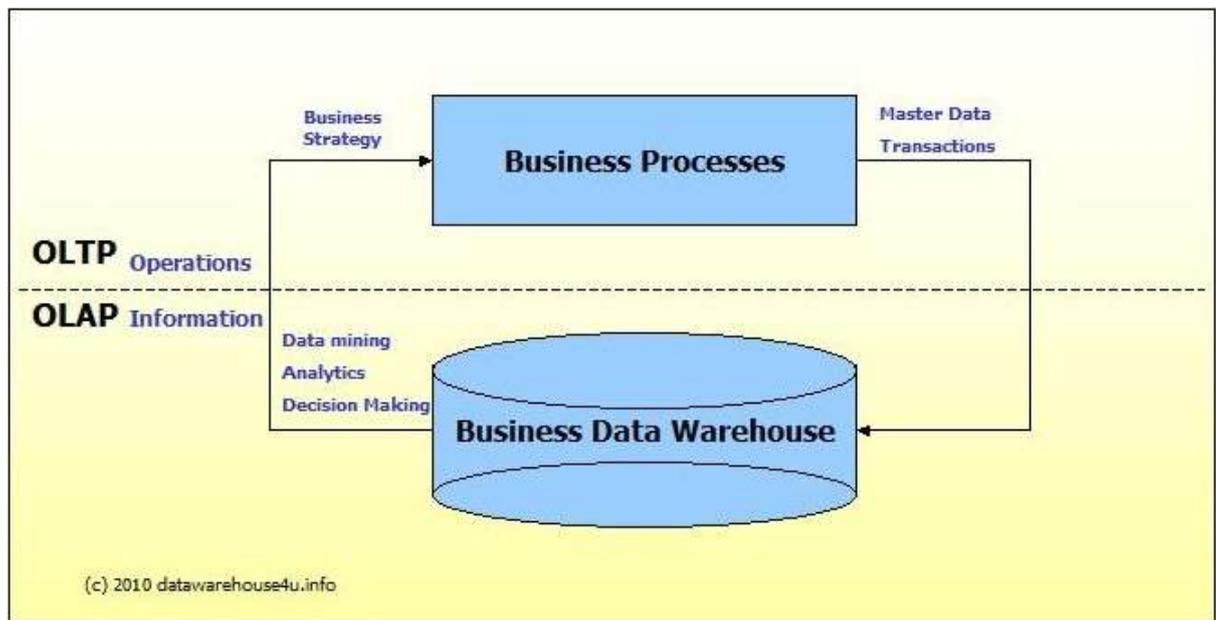
- The **data warehouse** on the other hand does not cater to real time operational requirements of the enterprise. It is more a storehouse of current and historical data and may also contain data extracted from external data sources.

Differences	
<b>Data warehouse database</b>	<b>OLTP database</b>
Designed for analysis of business measures by categories and attributes	Designed for real time business operations.
Optimized for bulk loads and large, complex, unpredictable queries that access many rows per table.	Optimized for a common set of transactions, usually adding or retrieving a single row at a time per table.
Loaded with consistent, valid data; requires no real time validation	Optimized for validation of incoming data during transactions; uses validation data tables.
Supports few concurrent users relative to OLTP	Supports thousands of concurrent users.

- However, the data warehouse supports OLTP system by providing a place for the latter to offload data as it accumulates and by providing services which would otherwise degrade the performance of the database.

## OLTP vs. OLAP

We can divide IT systems into transactional (OLTP) and analytical (OLAP). In general we can assume that OLTP systems provide source data to data warehouses, whereas OLAP systems help to analyze it.



- **OLTP (On-line Transaction Processing)** is characterized by a large number of short on-line transactions (INSERT, UPDATE, DELETE). The main emphasis for OLTP systems is put on very fast query processing, maintaining [data integrity](#) in multi-access environments and an effectiveness measured by number of transactions per second. In OLTP database there is detailed and current data, and schema used to store transactional databases is the entity model (usually 3NF).

- **OLAP (On-line Analytical Processing)** is characterized by relatively low volume of transactions. Queries are often very complex and involve aggregations. For OLAP systems a response time is an effectiveness measure. OLAP applications are widely used by [Data Mining](#) techniques. In OLAP database there is aggregated, historical data, stored in multi-dimensional schemas (usually star schema).

The following table summarizes the major differences between OLTP and OLAP system design.

	<b>OLTP System Online Transaction Processing (Operational System)</b>	<b>OLAP System Online Analytical Processing (Data Warehouse)</b>
Source of data	Operational data; OLTPs are the original source of the data.	Consolidation data; OLAP data comes from the various OLTP Databases
Purpose of data	To control and run fundamental business tasks	To help with planning, problem solving, and decision support
What the data	Reveals a snapshot of ongoing <a href="#">business processes</a>	Multi-dimensional views of various kinds of business activities
Inserts and Updates	Short and fast inserts and updates initiated by end users	Periodic long-running batch jobs refresh the data
Queries	Relatively standardized and simple queries Returning relatively few records	Often complex queries involving aggregations
Processing Speed	Typically very fast	Depends on the amount of data involved; batch data refreshes and complex queries may take many hours; query speed can be improved by creating indexes
Space Requirements	Can be relatively small if historical data is archived	Larger due to the existence of aggregation structures and history data; requires more indexes than OLTP
<a href="#">Database Design</a>	Highly normalized with many tables	Typically de-normalized with fewer tables; use of

		star and/or snowflake schemas
Backup and Recovery	Backup religiously; operational data is critical to run the business, data loss is likely to entail significant monetary loss and legal liability	Instead of regular backups, some environments may consider simply reloading the OLTP data as a recovery method

### DIFFERENCE BETWEEN OLAP & OLTP:-

OLTP	OLAP
1. Current data.	1. Current and historical data.
2. Short database transactions .	2. Long database transactions.
3. Short database transactions .	3. Batch update/insert/delete.
4. Normalization is promoted .	4. Denormalization is promoted .
5. High volume transactions .	5. Low volume transactions.
6. Transaction recovery is necessary.	6. Transaction recovery is not necessary.

### HOW OLTP WORKS :-

#### TRANSACTION PROCESSING SYSTEM:

A **Transaction Processing System** or **Transaction Processing Monitor** is a set of information which processes a data transaction in a database system that monitors *transaction programs* (a special kind of program). The essence of a transaction program is that it manages data that must be left in a consistent state. E.g. if an electronic payment is made, the amount must be both withdrawn from one account and added to the other; it cannot complete only one of those steps. Either both must occur, or neither. In case of a failure preventing transaction completion, the partially executed transaction must be '[rolled back](#)' by the TPS. While this type of integrity must be provided also for [batch transaction processing](#), it is particularly important for online processing: if e.g. an airline seat reservation system is accessed by multiple operators, after an empty seat inquiry,

the seat reservation data must be locked until the reservation is made, otherwise another user may get the impression a seat is still free while it is actually being booked at the time. Without proper transaction monitoring, double bookings may occur. Other transaction monitor functions include [deadlock](#) detection and resolution (deadlocks may be inevitable in certain cases of cross-dependence on data), and transaction logging (in 'journals') for 'forward recovery' in case of massive failures.

## **TYPES:-**

### **Contrasted with batch processing.....**

Batch processing is not transaction processing. Batch processing involves processing several transactions at the same time, and the results of each transaction are not immediately available when the transaction is being entered;<sup>[1]</sup> there is a time delay. transactions are accumulated for a certain period (say for day) where updates are made especially after work.

### **Real-time and batch processing**

There are a number of differences between **real-time** and **batch processing**. These are outlined below:

Each transaction in real-time processing is unique. It is not part of a group of transactions, even though those transactions are processed in the same manner. Transactions in real-time processing are stand-alone both in the entry to the system and also in the handling of output.

Real-time processing requires the master file to be available more often for updating and reference than batch processing. The database is not accessible all of the time for batch processing.

Real-time processing has fewer errors than batch processing, as transaction data is validated and entered immediately. With batch processing, the data is organised and stored before the master file is updated. Errors can occur during these steps.

Infrequent errors may occur in real-time processing; however, they are often tolerated. It is not practical to shut down the system for infrequent errors.

More computer operators are required in real-time processing, as the operations are not centralised. It is more difficult to maintain a real-time processing system than a batch processing system.

## **Features**

### **Rapid response**

Fast performance with a rapid response time is critical. Businesses cannot afford to have customers waiting for a TPS to respond, the turnaround time from the input of the transaction to the production for the output must be a few seconds or less.

### **Reliability**

Many organizations rely heavily on their TPS; a breakdown will disrupt operations or even stop the business. For a TPS to be effective its failure rate must be very low. If a TPS does fail, then quick and accurate recovery must be possible. This makes well-designed backup and recovery procedures essential.

### **Inflexibility**

A TPS wants every transaction to be processed in the same way regardless of the user, the customer or the time for day. If a TPS were flexible, there would be too many opportunities for non-standard operations, for example, a commercial airline needs to consistently accept airline reservations from a range of travel agents, accepting different transactions data from different travel agents would be a problem.

### **Controlled processing**

The processing in a TPS must support an organization's operations. For example if an organization allocates roles and responsibilities to

particular employees, then the TPS should enforce and maintain this requirement.

Example : ATM Transaction

## **ACID test properties: first definition**

### **Atomicity**

Main article: [Atomicity \(database systems\)](#)

A transaction's changes to the state are atomic: either all happen or none happen. These changes include database changes, messages, and actions on transducers.<sup>[2]</sup>

### **Consistency**

A transaction is a correct transformation of the state. The actions taken as a group do not violate any of the integrity constraints associated with the state. This requires that the transaction be a correct program!<sup>[2]</sup>

### **Isolation**

Even though transactions execute concurrently, it appears to each transaction T, that others executed either before T or after T, but not both.<sup>[2]</sup>

### **Durability**

Once a transaction completes successfully (commits), its changes to the state survive failures.<sup>[2]</sup>

### **Concurrency**

Ensures that two users cannot change the same data at the same time. That is, one user cannot change a piece of data before another user has finished with it. For example, if an airline ticket agent starts to reserve the last seat on a flight, then another agent cannot tell another passenger that a seat is available.

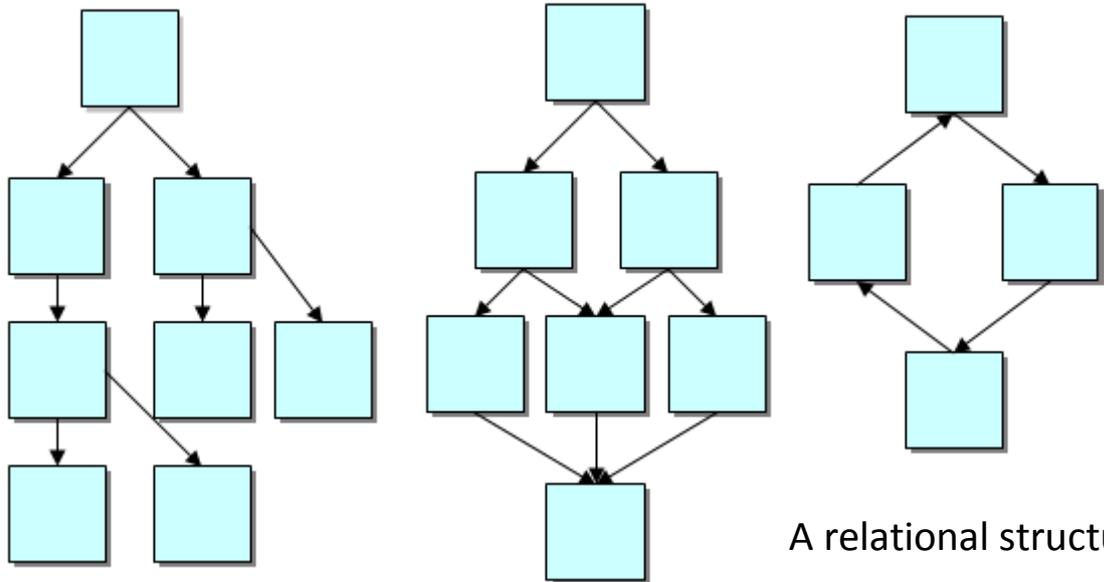
## **Storing and retrieving**

Storing and retrieving information from a TPS must be efficient and effective. The data are stored in warehouses or other databases, the system must be well designed for its backup and recovery procedures.

## **Databases and files**

The storage and retrieval of data must be accurate as it is used many times throughout the day. A database is a collection of data neatly organized, which stores the accounting and operational records in the database. Databases are always protective of their delicate data, so they usually have a restricted view of certain data. Databases are designed using hierarchical, network or relational structures; each structure is effective in its own sense.

- Hierarchical structure: organizes data in a series of levels, hence why it is called hierarchal. Its top to bottom like structure consists of nodes and branches; each child node has branches and is only linked to one higher level parent node.
- Network structure: Similar to hierarchical, network structures also organizes data using nodes and branches. But, unlike hierarchical, each child node can be linked to multiple, higher parent nodes.
- Relational structure: Unlike network and hierarchical, a relational database organizes its data in a series of related tables. This gives flexibility as relationships between the tables are built.



A hierarchical structure.

A network structure.

A relational structure.

The following features are included in real time transaction processing systems:

- **Good data placement:** The database should be designed to access patterns of data from many simultaneous users.
- **Short transactions:** Short transactions enables quick processing. This avoids concurrency and paces the systems.
- **Real-time backup:** Backup should be scheduled between low times of activity to prevent lag of the server.
- **High normalization:** This lowers redundant information to increase the speed and improve concurrency, this also improves backups.
- **Archiving of historical data:** Uncommonly used data are moved into other databases or backed up tables. This keeps tables small and also improves backup times.
- **Good hardware configuration:** Hardware must be able to handle many users and provide quick response times.

In a TPS, there are 5 different types of files. The TPS uses the files to store and organize its transaction data:

- **Master file:** Contains information about an organization's business situation. Most transactions and databases are stored in the master file.
- **Transaction file:** It is the collection of transaction records. It helps to update the master file and also serves as audit trails and transaction history.
- **Report file:** Contains data that has been formatted for presentation to a user.
- **Work file:** Temporary files in the system used during the processing.
- **Program file:** Contains the instructions for the processing of data.

## Data warehouse

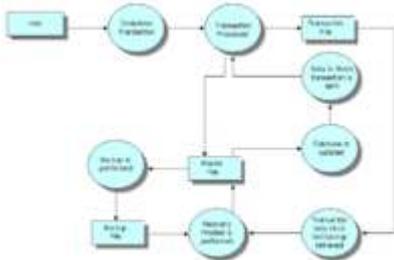
Main article: [Data warehouse](#)

A data warehouse is a database that collects information from different sources. When it's gathered in real-time transactions it can be used for analysis efficiently if it's stored in a data warehouse. It provides data that are *consolidated*, *subject-oriented*, *historical* and *read-only*:

- **Consolidated:** Data are organised with consistent naming conventions, measurements, attributes and semantics. It allows data from a data warehouse from across the organization to be effectively used in a consistent manner.
- **Subject-oriented:** Large amounts of data are stored across an organization, some data could be irrelevant for reports and makes querying the data difficult. It organizes only key business information from operational sources so that it's available for analysis.
- **Historical:** Real-time TPS represent the current value at any time, an example could be stock levels. If past data are kept, querying the database could return a different response. It stores series of snapshots for an organisation's operational data generated over a period of time.

- **Read-only:** Once data are moved into a data warehouse, it becomes read-only, unless it was incorrect. Since it represents a snapshot of a certain time, it must never be updated. Only operations which occur in a data warehouse are loading and querying data.

## Backup procedures



A Dataflow Diagram of backup and recovery procedures.

Since business organizations have become very dependent on TPSs, a breakdown in their TPS may stop the business' regular routines and thus stopping its operation for a certain amount of time. In order to prevent data loss and minimize disruptions when a TPS breaks down a well-designed backup and recovery procedure is put into use. The recovery process can rebuild the system when it goes down.

## Recovery process

A TPS may fail for many reasons. These reasons could include a system failure, human errors, hardware failure, incorrect or invalid data, computer viruses, software application errors or natural or man-made disasters. As it's not possible to prevent all TPS failures, a TPS must be able to cope with failures. The TPS must be able to detect and correct errors when they occur. A TPS will go through a recovery of the database to cope when the system fails, it involves the backup, journal, checkpoint, and recovery manager:

- **Journal:** A journal maintains an audit trail of transactions and database changes. Transaction logs and Database change logs are used, a transaction log records all the essential data for

each transactions, including data values, time of transaction and terminal number. A database change log contains before and after copies of records that have been modified by transactions.

- Checkpoint: A checkpoint record contains necessary information to restart the system. These should be taken frequently, such as several times an hour. It is possible to resume processing from the most-recent checkpoint when a failure occurs with only a few minutes of processing work that needs to be repeated.
- Recovery Manager: A recovery manager is a program which restores the database to a correct condition which can restart the transaction processing.

Depending on how the system failed, there can be two different recovery procedures used. Generally, the procedures involves restoring data that has been collected from a backup device and then running the transaction processing again. Two types of recovery are *backward recovery* and *forward recovery*:

- Backward recovery: used to undo unwanted changes to the database. It reverses the changes made by transactions which have been aborted. It involves the logic of reprocessing each transaction, which is very time-consuming.
- Forward recovery: it starts with a backup copy of the database. The transaction will then reprocess according to the transaction journal that occurred between the time the backup was made and the present time. It's much faster and more accurate.

### **Types of back-up procedures**

There are two main types of Back-up Procedures: **Grandfather-father-son** and **Partial backups**:

## Grandfather-father-son

This procedure refers to at least three generations of backup master files. thus, the most recent backup is the son, the oldest backup is the grandfather. It's commonly used for a *batch transaction processing system* with a magnetic tape. If the system fails during a batch run, the master file is recreated by using the son backup and then restarting the batch. However if the son backup fails, is corrupted or destroyed, then the next generation up backup (father) is required. Likewise, if that fails, then the next generation up backup (grandfather) is required. Of course the older the generation, the more the data may be out of date. Organizations can have up to twenty generations of backup.

## Partial backups

This only occurs when parts of the master file are backed up. The master file is usually backed up to magnetic tape at regular times, this could be daily, weekly or monthly. Completed transactions since the last backup are stored separately and are called **journals**, or **journal files**. The master file can be recreated from the journal files on the backup tape if the system is to fail.

## Updating in a batch

This is used when transactions are recorded on paper (such as bills and invoices) or when it's being stored on a magnetic tape. Transactions will be collected and updated as a batch at when it's convenient or economical to process them. Historically, this was the most common method as the information technology did not exist to allow real-time processing.

The two stages in batch processing are:

- Collecting and storage of the transaction data into a transaction file - this involves sorting the data into sequential order.
- Processing the data by updating the master file - which can be difficult, this may involve data additions, updates and deletions

that may require to happen in a certain order. If an error occurs, then the entire batch fails.

Updating in batch requires sequential access - since it uses a magnetic tape this is the only way to access data. A batch will start at the beginning of the tape, then reading it from the order it was stored; it's very time-consuming to locate specific transactions.

The information technology used includes a secondary storage medium which can store large quantities of data inexpensively (thus the common choice of a magnetic tape). The software used to collect data does not have to be online - it doesn't even need a user interface.

### **Updating in real-time**

This is the immediate processing of data. It provides instant confirmation of a transaction. This involves a large amount of users who are simultaneously performing transactions to change data. Because of advances in technology (such as the increase in the speed of data transmission and larger bandwidth), real-time updating is now possible.

Steps in a real-time update involve the sending of a transaction data to an online database in a master file. The person providing information is usually able to help with error correction and receives confirmation of the transaction completion.

Updating in real-time uses direct access of data. This occurs when data are accessed without accessing previous data items. The storage device stores data in a particular location based on a mathematical procedure. This will then be calculated to find an approximate location of the data. If data are not found at this location, it will search through successive locations until it's found.

The information technology used could be a secondary storage medium that can store large amounts of data and provide quick access (thus the common choice of a magnetic disk). It requires a user-friendly interface as it's important for rapid response time.

**Reservation Systems** Reservation systems are used for any type of business where a service or a product is set aside for a customer to use for a future time.

## **MERCHANT PAYMENT SYSTEM:-**

-Electronic payment systems exist in a variety of forms which can be divided into two groups: wholesale payment systems and retail payment systems. Wholesale payment systems exist for nonconsumer transactions-transactions initiated among and between banks, corporations, governments, and other financial service firms. High-value wholesale payments flow through the three major interbank funds transfer systems: the Clearing House Interbank Payment Systems (CHIPS),<sup>6</sup> the Society for Worldwide Interbank Financial Telecommunications (SWIFT),<sup>7</sup> and Fedwire.<sup>8</sup> Electronic transfers utilizing these types of payment systems are beyond the scope of this Note.

Retail electronic payment systems encompass those transactions involving consumers. These transactions involve the use of such payment mechanisms as credit cards, automated teller machines (ATMs), debit cards, point-of-sale (POS) terminals, home banking, and telephone bill-paying services. Payments for these mechanisms are conducted online and flow through the check truncation system<sup>9</sup> and the ACH.<sup>10</sup> Electronic transfers involving these types of payment mechanisms and payment systems are also beyond the scope of this Note.

The distinction between wholesale and retail electronic payment systems parallels the distinction that has evolved in regulating these systems. Wholesale electronic payment systems are regulated by Article 4A of the Uniform Commercial Code. Retail electronic payment systems are regulated by the Consumer Credit Protection Act;<sup>11</sup> the Truth in Lending Act<sup>12</sup> and its adjunct, Regulation Z;<sup>13</sup> and the Electronic Funds Transfer Act (EFTA)<sup>14</sup> and its adjunct, Part 205 of Regulation E.<sup>15</sup> This regulatory distinction reflects the kinds of parties involved-rules for retail electronic payment systems are fashioned with the consumer in mind whereas rules for wholesale payment systems are fashioned with commercial parties in mind.

-The term '**electronic payment**' is a collective phrase for the many different kinds of electronic payment methods available (also meaning online payment), and the processing of transactions and their application within online merchants and ecommerce websites.

It is essential for all online businesses to be able to accept and process electronic payments in a fast and secure way. Businesses can gain a significant advantage over their competitors by providing an instant electronic payment service as it lets customers pay by their preferred way of credit or debit card.

Electronic payments systems can also increase your cash flow, reduce administrative costs and labour and provide yet another way for your customers to pay. Care must be taken when choosing an electronic payment solution as it will need to fit within the constraints of your particular online business and integrate seamlessly within your website.