# Data Model XML

Tushar B. Kute,
http://tusharkute.com

# What is XML?

- Xml (eXtensible Markup Language) is a mark up language.
- XML is designed to store and transport data.
- Xml was released in late 90's. it was created to provide an easy to use and store self describing data.
- XML became a W3C Recommendation on February 10, 1998.
- XML is not a replacement for HTML.
- XML is designed to be self-descriptive.
- XML is designed to carry data, not to display data.
- XML tags are not predefined. You must define your own tags.
- XML is platform independent and language independent.

# Why XML?

- Platform Independent and Language Independent: The main benefit of xml is that you can use it to take data from a program like Microsoft SQL, convert it into XML then share that XML with other programs and platforms.
- The main thing which makes XML truly powerful is its international acceptance. Many corporation use XML interfaces for databases, programming, office application mobile phones and more. It is due to its platform independent feature.

# XML Database

- XML database is a data persistence software system used for storing the huge amount of information in XML format.
- It provides a secure place to store XML documents.
- You can query your stored data by using XQuery, export and serialize into desired format.
- XML databases are usually associated with document-oriented databases.

# XML Database

- Types of XML databases
- There are two types of XML databases.
  - XML-enabled database
  - Native XML database (NXD)

# XML Database

- XML-enable Database
  - XML-enable database works just like a relational database. It is like an extension provided for the conversion of XML documents. In this database, data is stored in table, in the form of rows and columns.
- Native XML Database
  - Native XML database is used to store large amount of data. Instead of table format, Native XML database is based on container format. You can query data by XPath expressions.
  - Native XML database is preferred over XML-enable database because it is highly capable to store, maintain and query XML documents.

# XML Database

```
<?xml version="1.0"?>
<contact-info>
  <contact1>
    <name>Tushar B. Kute</name>
    <company>mituresearch.in</company>
    <phone>020 67356627</phone>
  </contact1>
  <contact2>
    <name>Rashmi Thorave </name>
    <company>Prakrut Prakashan</company>
    <phone>09990449935</phone>
  </contact2>
</contact-info>
```

# Features

- XML separates data from HTML
  - If you need to display dynamic data in your HTML document, it will take a lot of work to edit the HTML each time the data changes.
- XML simplifies data sharing
  - In the real world, computer systems and databases contain data in incompatible formats.
  - XML data is stored in plain text format. This provides a software- and hardware-independent way of storing data.

# Features

- XML simplifies data transport
  - One of the most time-consuming challenges for developers is to exchange data between incompatible systems over the Internet.
- XML simplifies Platform change
  - Upgrading to new systems (hardware or software platforms), is always time consuming. Large amounts of data must be converted and incompatible data is often lost.

# Features

- XML increases data availability
  - Different applications can access your data, not only in HTML pages, but also from XML data sources.
- XML can be used to create new internet languages
  - A lot of new Internet languages are created with XML.
  - Here are some examples:
    - XHTML, WSDL, WML

# Querying and Transformation

- Translation of information from
  - One XML schema to another
  - Querying on XML data
- Above two are closely related, and handled by the same tools
- Standard XML querying/translation languages
  - XPath
    - Simple language consisting of path expressions
  - XSLT
    - Simple language designed for translation from XML to XML and XML to HTML
  - XQuery
    - An XML query language with a rich set of features

# Querying and Transformation

- Query and transformation languages are based on a tree model of XML data
- An XML document is modeled as a tree, with nodes corresponding to elements and attributes
  - Element nodes have child nodes, which can be attributes or subelements
  - Text in an element is modeled as a text node child of the element
  - Children of a node are ordered according to their order in the XML document
  - Element and attribute nodes (except for the root node) have a single parent, which is an element node
  - The root node has a single child, which is the root element of the document

# XPath

- XPath is used to address (select) parts of documents using path expressions
- A path expression is a sequence of steps separated by "/"
  - Think of file names in a directory hierarchy
- Result of path expression: set of values that along with their containing elements/attributes match the specified path
- E.g. /bank-2/customer/customer_name evaluated on the bank-2 data we saw earlier returns
  - <customer_name>Joe</customer_name>
  - <customer_name>Mary</customer_name>
- E.g. /bank-2/customer/customer_name/text( ) returns the same names, but without the enclosing tags

# XPath

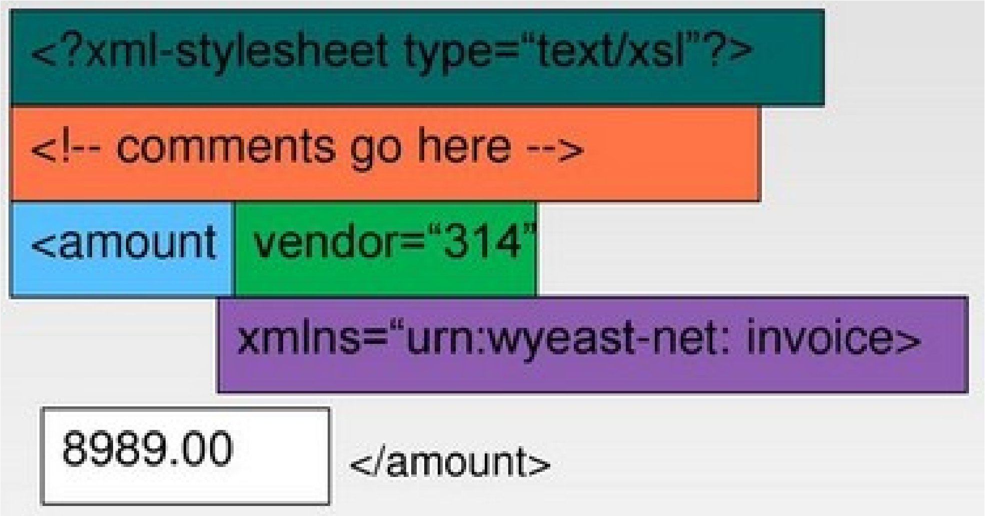
- The initial “/” denotes root of the document (above the top-level tag)
- Path expressions are evaluated left to right
  - Each step operates on the set of instances produced by the previous step
- Selection predicates may follow any step in a path, in [ ]
  - E.g. /bank-2/account[balance > 400]
    - returns account elements with a balance value greater than 400
    - /bank-2/account[balance] returns account elements containing a balance subelement
- Attributes are accessed using
  - E.g. /bank-2/account[balance > returns the account numbers of accounts with balance > 400
  - IDREF attributes are not dereferenced automatically

# Traversing the source tree

- Templates definitions are applied to elements of source document
- First Template matching a pattern chosen
- Pattern
  - subset of XPath expressions
  - can match elements and attributes
  - can use node tests and predicates
- XPath
  - Simple language to identify parts of an XML document
  - Similar to paths in a file system

# Traversing the source tree

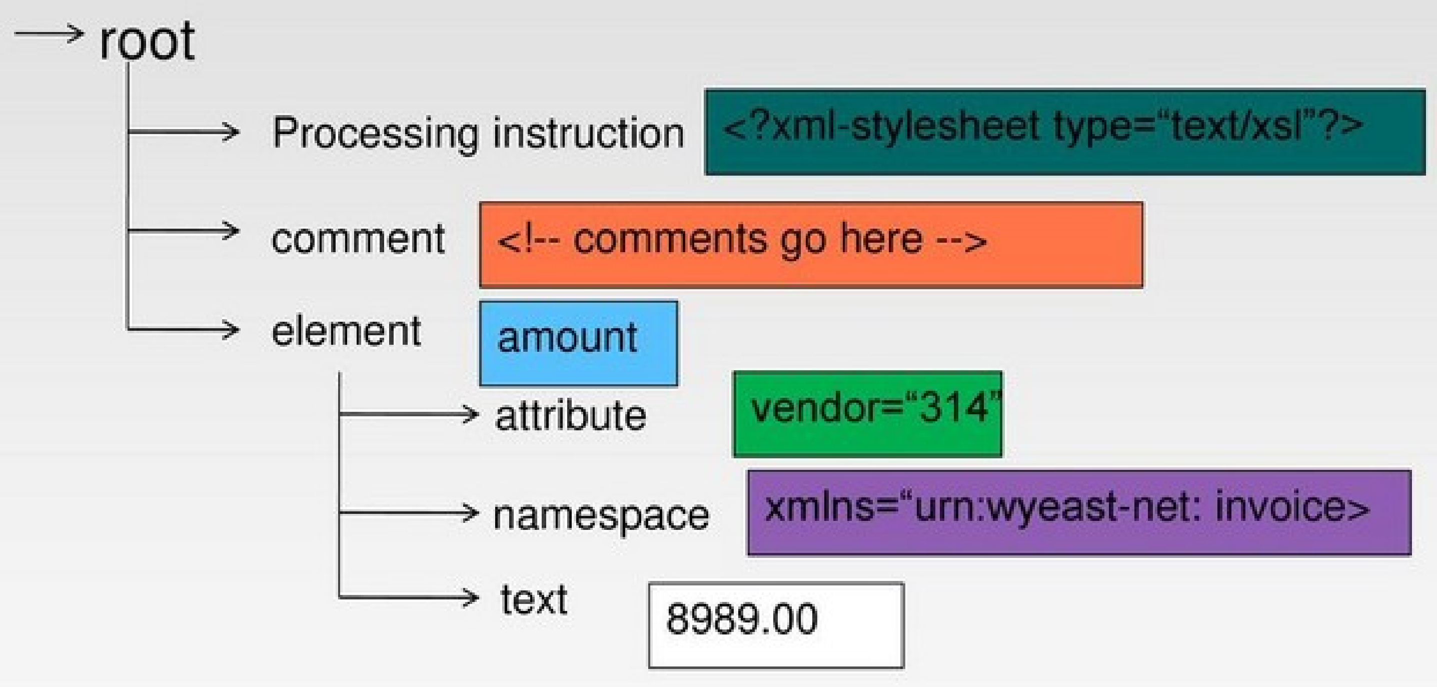
- The XPath data model
- XML documents consists of seven types
  - Root node
  - Elements
  - Attribute
  - Text
  - Namespace
  - Comment
  - Processing
  - instruction



```
<?xml-stylesheet type=“text/xsl”?>
<!-- comments go here -->
<amount vendor=“314”
xmlns=“urn:wyeast-net: invoice>
8989.00 </amount>
```

# Traversing the source tree

- XML structured can viewed as tree with different types of nodes



root

Processing instruction <?xml-stylesheet type=“text/xsl”?>

comment <!-- comments go here -->

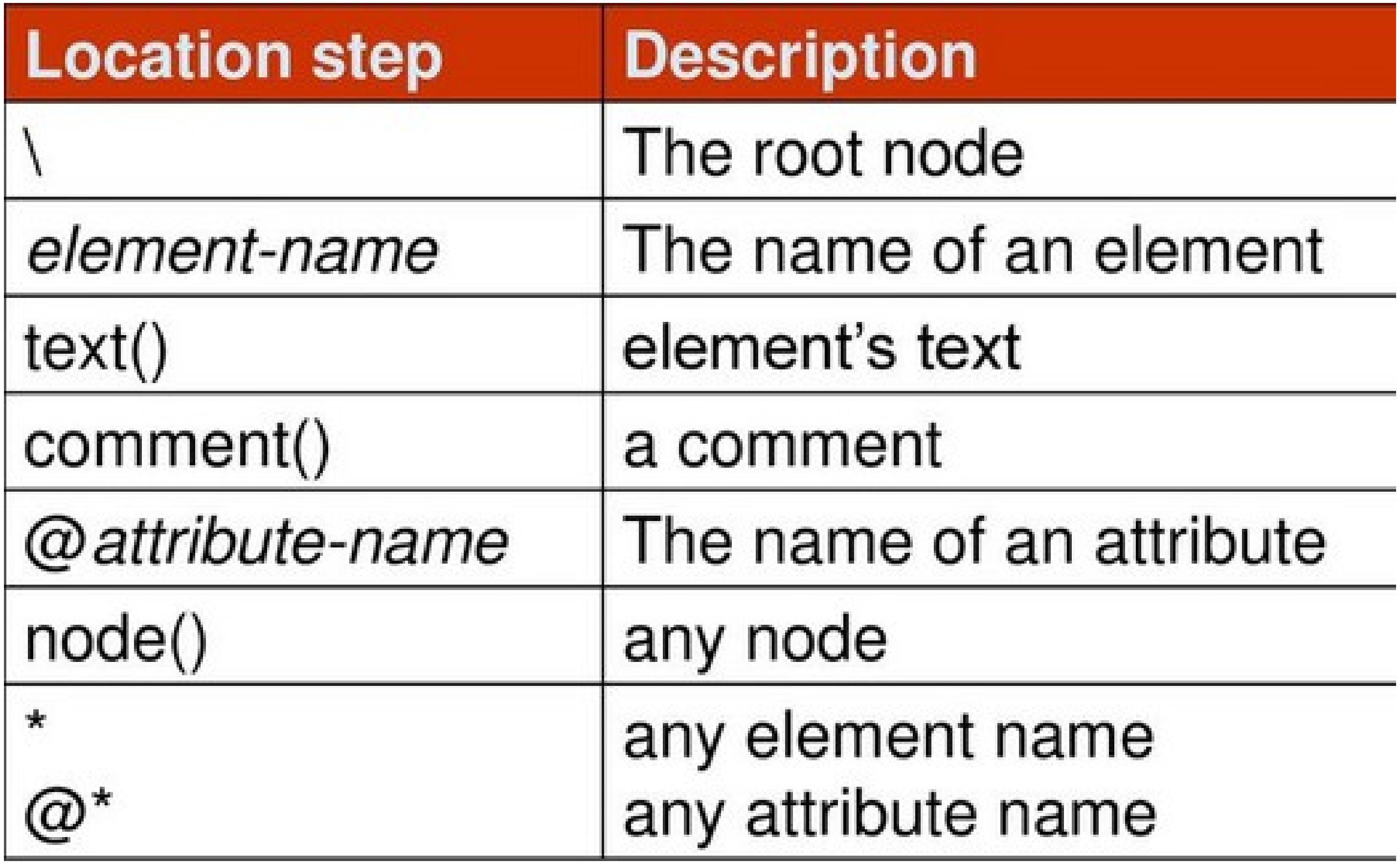
element amount

attribute vendor=“314”

namespace xmlns=“urn:wyeast-net: invoice>

text 8989.00

# XPath Data Model



| Location step | Description |
|---|---|
| \ | The root node |
| *element-name* | The name of an element |
| text() | element’s text |
| comment() | a comment |
| @*attribute-name* | The name of an attribute |
| node() | any node |
| *<br>@* | any element name<br>any attribute name |

# XPath data model

- Result of an XPath location: duplicate free set of nodes
  - /movies/movie: the set of all movie elements
  - /movies/movie/rating: the set of all rating elements “<rating>3</rating>”, “<rating>2</rating>”
  - /movies/movie/title/@lang the set of all language attribute names
  - <movies>
    - <movie>
      - <title>Man In Black</title>
      - <rating>3</rating>
    - </movie>
    - <movie>
      - <title lang=“en”>Batman</title>
      - <title lang=“ko”>Bateman</title>
      - <rating>2</rating>
    - </movie>
  - </movies>

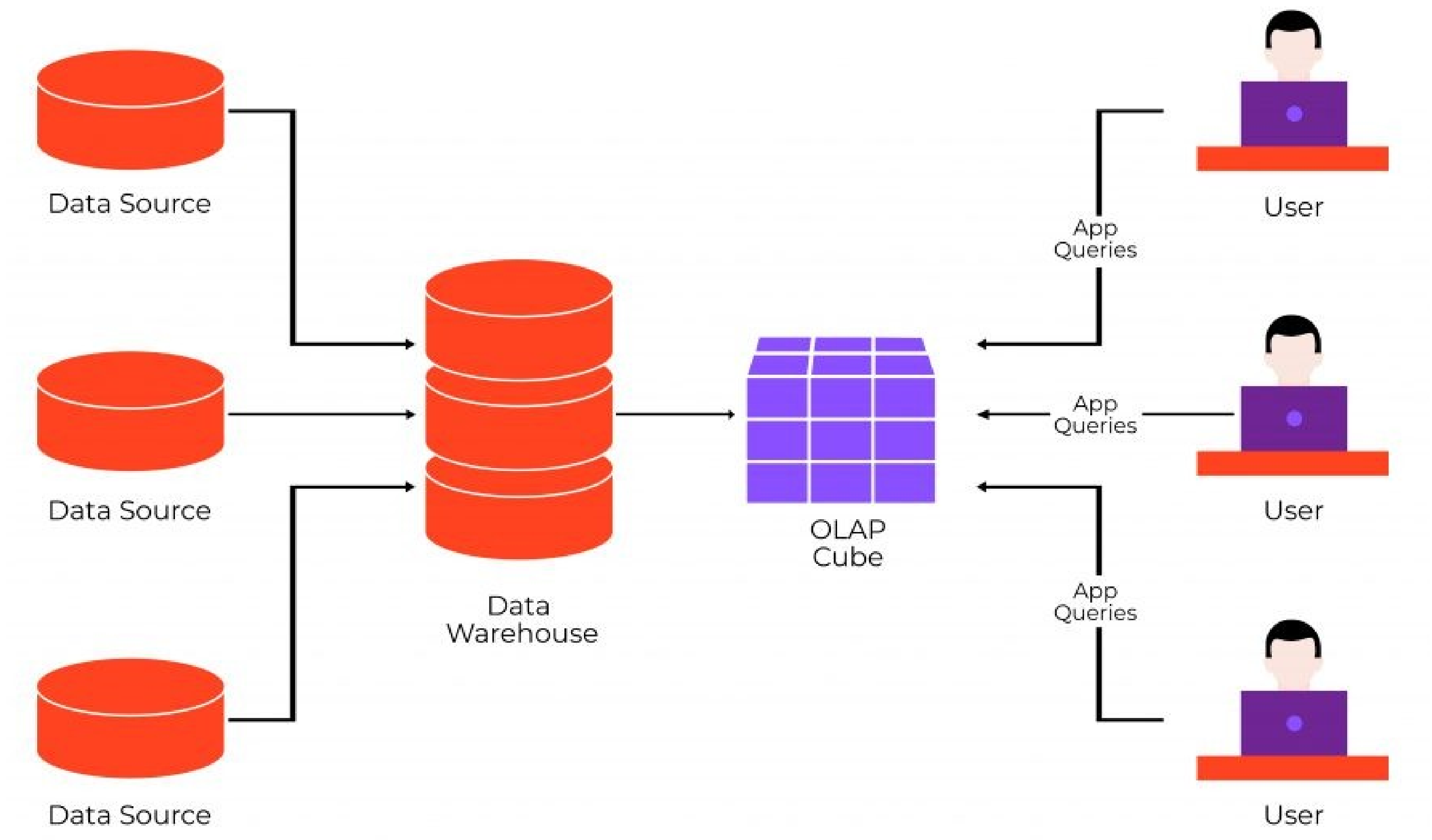
# XQuery

- XQuery is a general purpose query language for XML data
- Currently being standardized by the World Wide Web Consortium (W3C)
  - The textbook description is based on a January 2005 draft of the standard. The final version may differ, but major features likely to stay unchanged.
- XQuery is derived from the Quilt query language, which itself borrows from SQL, XQL and XML-QL
- XQuery uses a
  - for … let … where … order by …result …
  - syntax
    - for □ SQL from where □ SQL where order by □ SQL order by result □ SQL select let allows temporary variables, and has no equivalent in SQL

# OLAP

- Online Analytical Processing, a category of software tools which provide analysis of data for business decisions.
- OLAP systems allow users to analyze database information from multiple database systems at one time.
- The primary objective is data analysis and not data processing.

# OLAP


Data Source
Data Source
Data Source
Data
Warehouse
OLAP
Cube
App
Queries
App
Queries
App
Queries
User
User
User

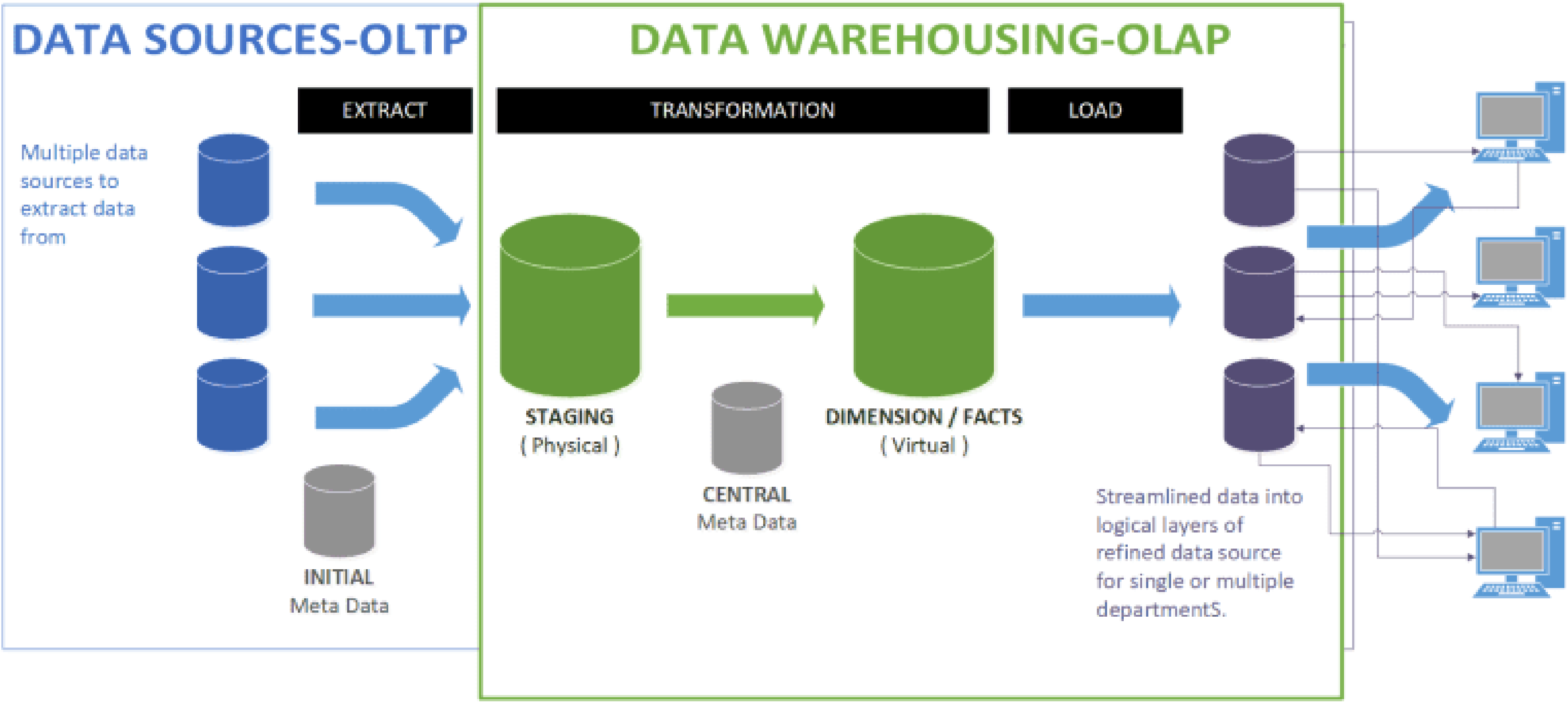
# OLAP: Examples

- Any Datawarehouse system is an OLAP system. Uses of OLAP are as follows
  - A company might compare their mobile phone sales in September with sales in October, then compare those results with another location which may be stored in a sperate database.
  - Amazon analyzes purchases by its customers to come up with a personalized homepage with products which likely interest to their customer.

# OLTP

- Online transaction processing shortly known as OLTP supports transaction-oriented applications in a 3-tier architecture. OLTP administers day to day transaction of an organization.
- The primary objective is data processing and not data analysis

# OLTP


DATA SOURCES-OLTP
DATA WAREHOUSING-OLAP
EXTRACT
TRANSFORMATION
LOAD
Multiple data sources to extract data from
STAGING
( Physical )
DIMENSION / FACTS
( Virtual )
CENTRAL
Meta Data
INITIAL
Meta Data
Streamlined data into logical layers of refined data source for single or multiple departmentS.

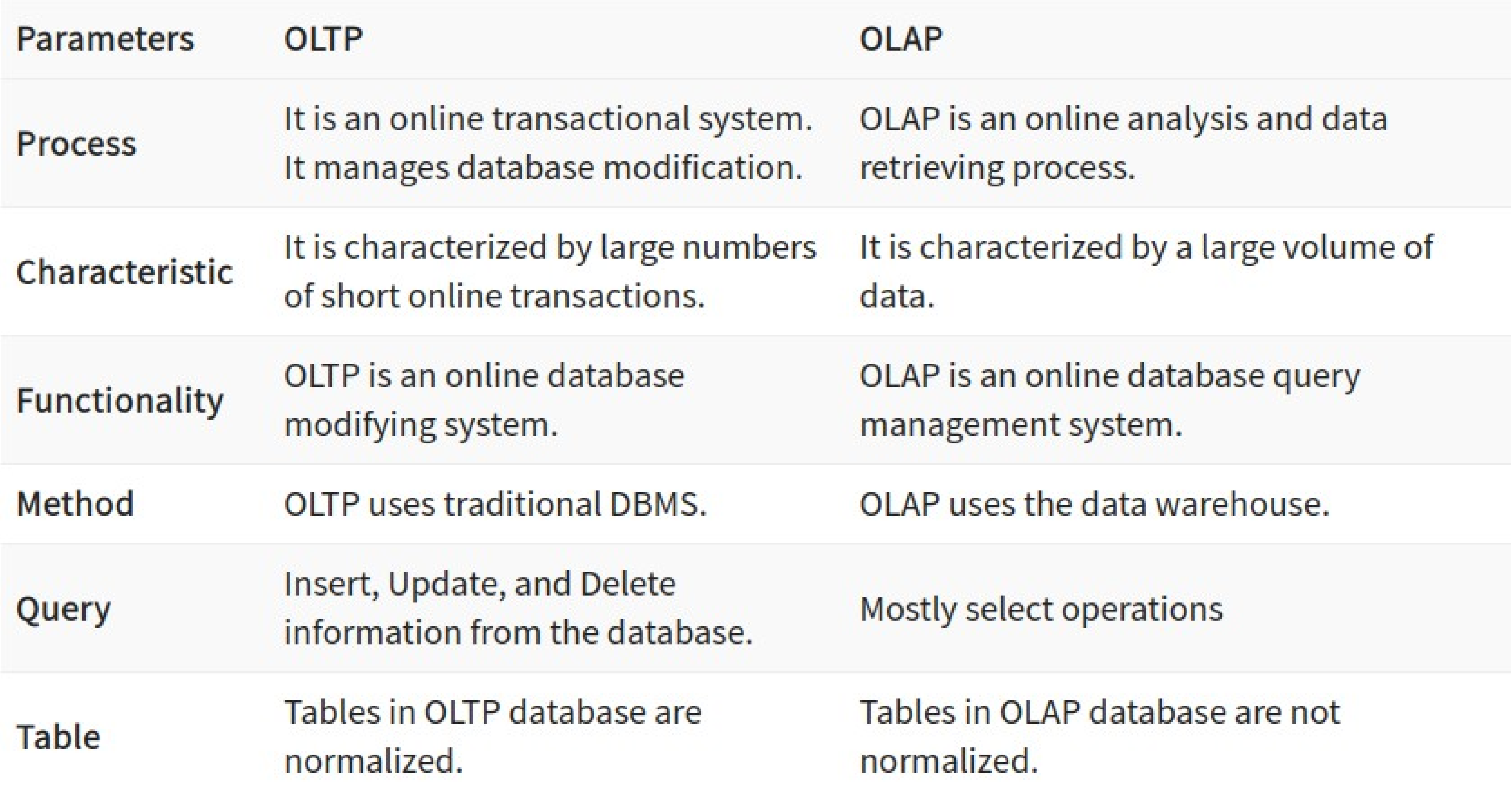
# OLTP: Examples

- An example of OLTP system is ATM center.
- Assume that a couple has a joint account with a bank. One day both simultaneously reach different ATM centers at precisely the same time and want to withdraw total amount present in their bank account.
- However, the person that completes authentication process first will be able to get money. In this case, OLTP system makes sure that withdrawn amount will be never more than the amount present in the bank.
- The key to note here is that OLTP systems are optimized for transactional superiority instead data analysis.

# OLTP: Examples

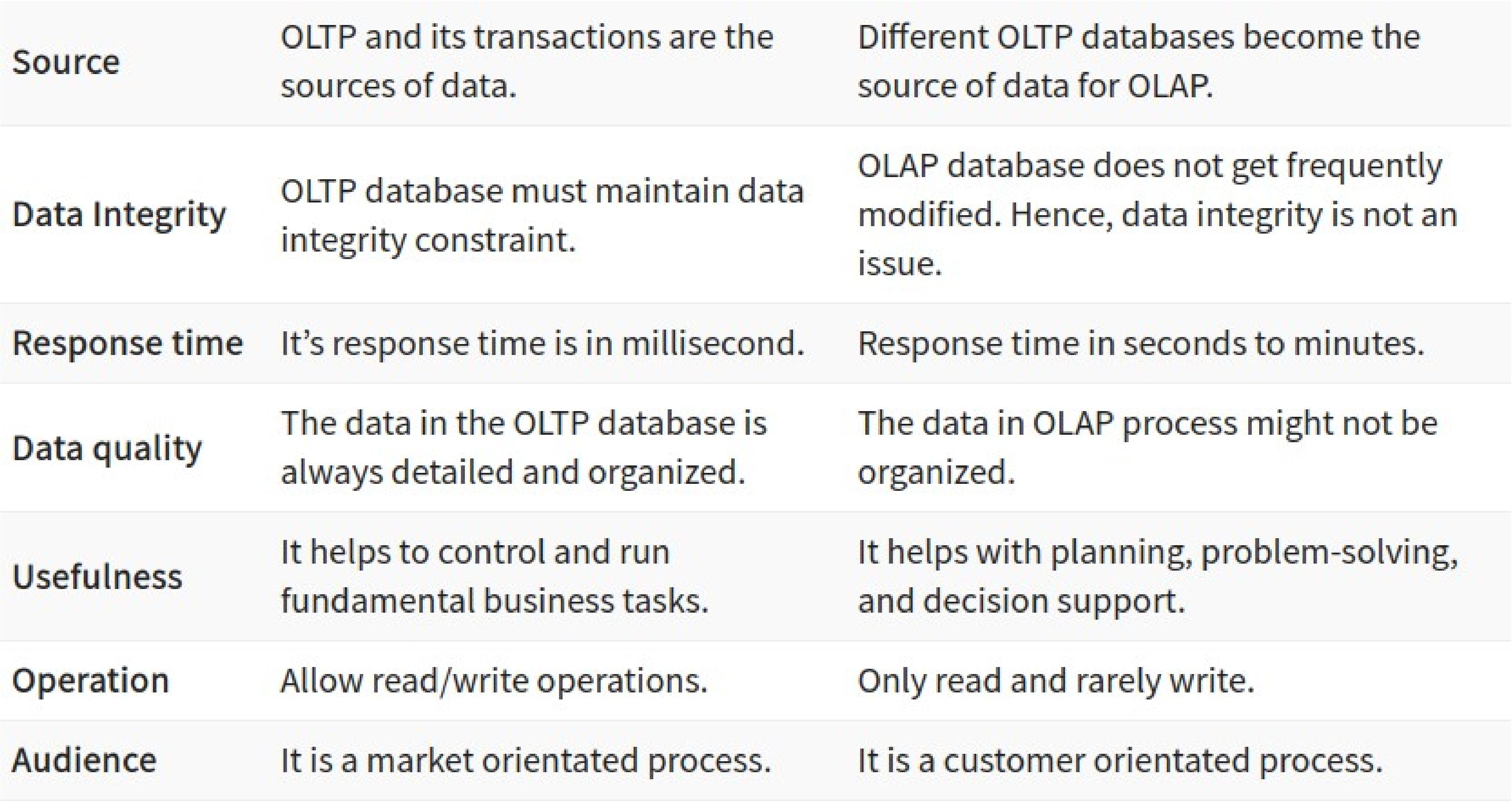
- Other examples of OLTP applications are:
  - Online banking
  - Online airline ticket booking
  - Sending a text message
  - Order entry
  - Add a book to shopping cart

# OLTP vs. OLAP



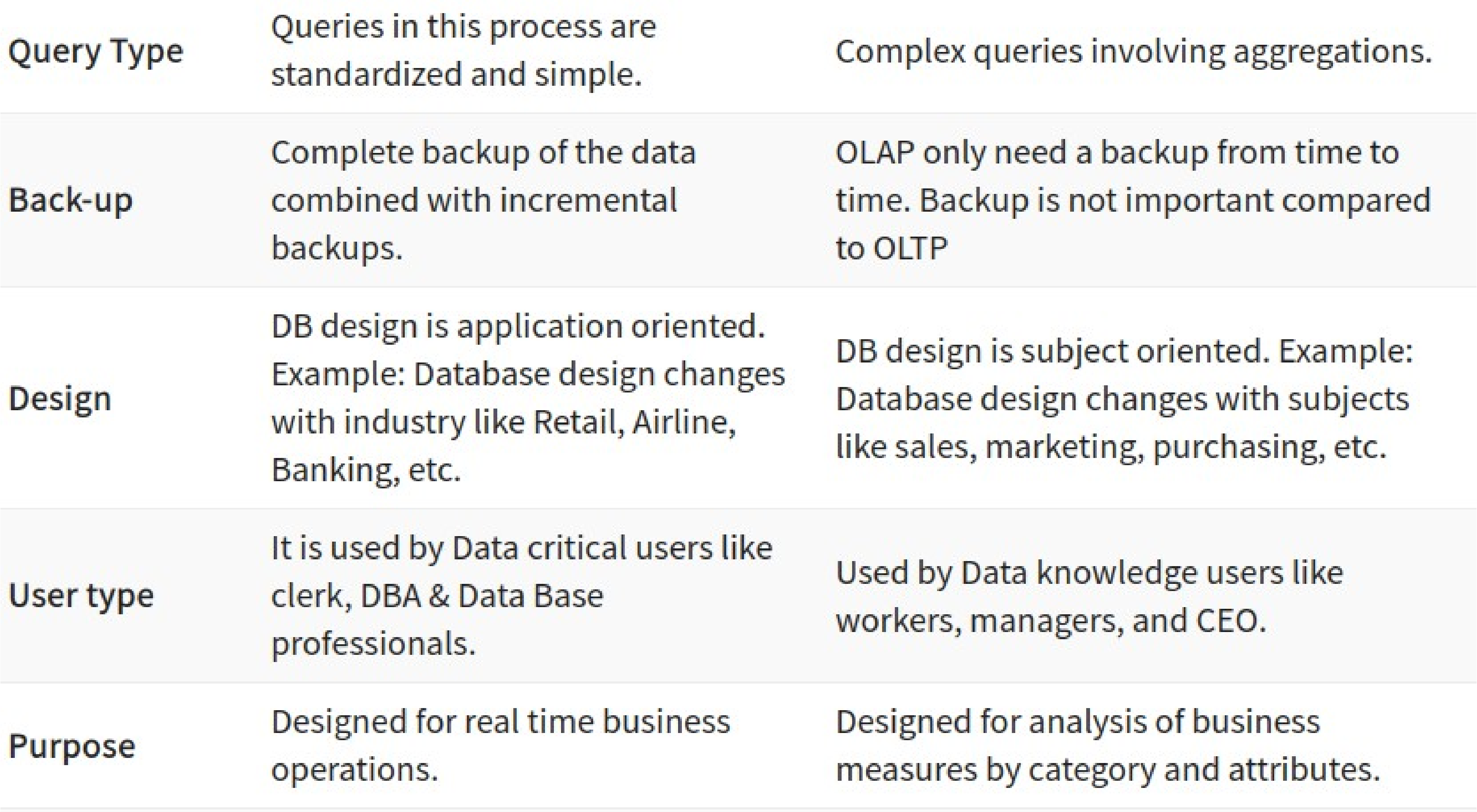
| Parameters | OLTP | OLAP |
|---|---|---|
| Process | It is an online transactional system. It manages database modification. | OLAP is an online analysis and data retrieving process. |
| Characteristic | It is characterized by large numbers of short online transactions. | It is characterized by a large volume of data. |
| Functionality | OLTP is an online database modifying system. | OLAP is an online database query management system. |
| Method | OLTP uses traditional DBMS. | OLAP uses the data warehouse. |
| Query | Insert, Update, and Delete information from the database. | Mostly select operations |
| Table | Tables in OLTP database are normalized. | Tables in OLAP database are not normalized. |

# OLTP vs. OLAP



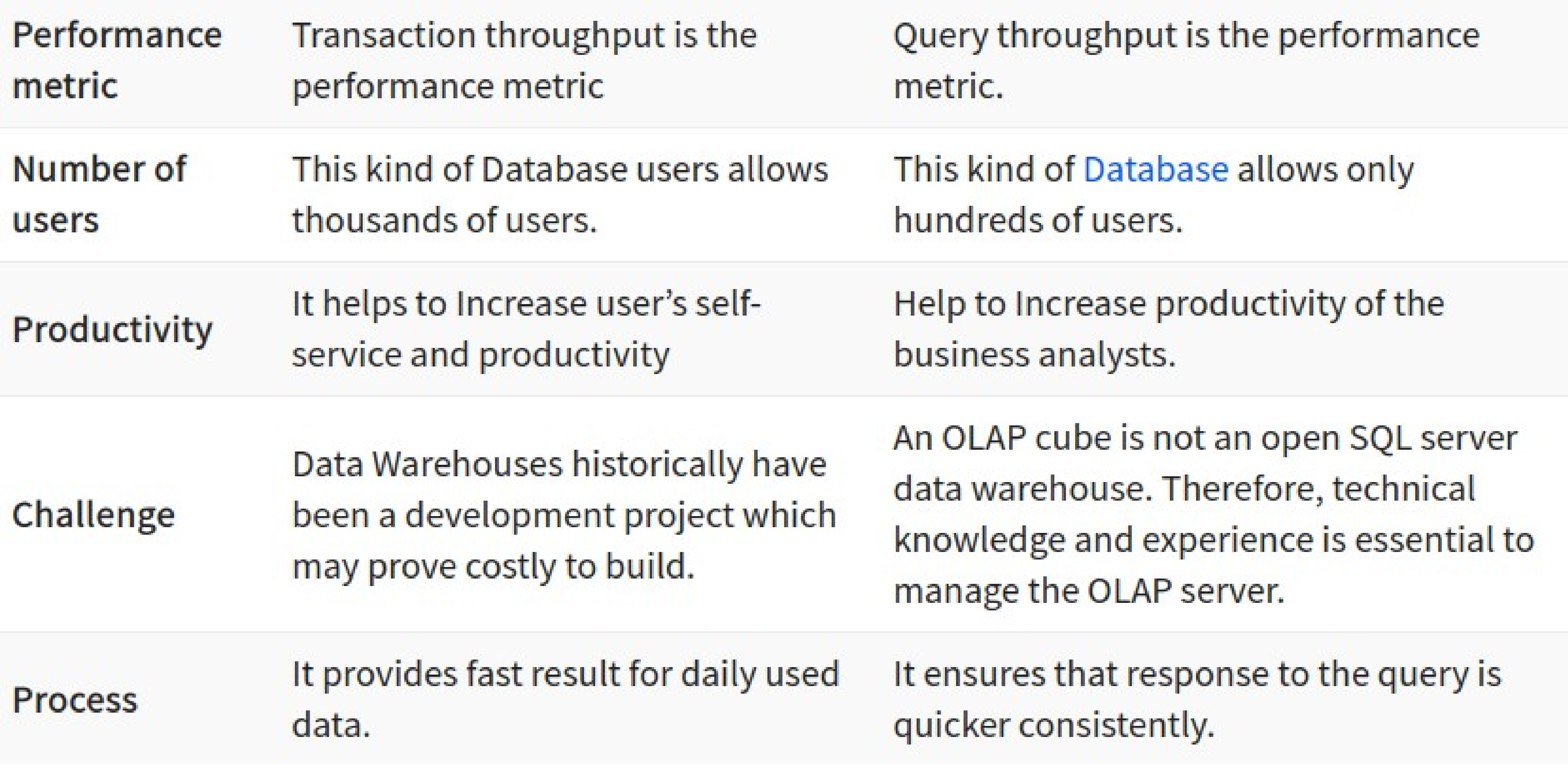
| | | |
|---|---|---|
| **Source** | OLTP and its transactions are the sources of data. | Different OLTP databases become the source of data for OLAP. |
| **Data Integrity** | OLTP database must maintain data integrity constraint. | OLAP database does not get frequently modified. Hence, data integrity is not an issue. |
| **Response time** | It's response time is in millisecond. | Response time in seconds to minutes. |
| **Data quality** | The data in the OLTP database is always detailed and organized. | The data in OLAP process might not be organized. |
| **Usefulness** | It helps to control and run fundamental business tasks. | It helps with planning, problem-solving, and decision support. |
| **Operation** | Allow read/write operations. | Only read and rarely write. |
| **Audience** | It is a market orientated process. | It is a customer orientated process. |

# OLTP vs. OLAP



| | | |
|---|---|---|
| **Query Type** | Queries in this process are standardized and simple. | Complex queries involving aggregations. |
| **Back-up** | Complete backup of the data combined with incremental backups. | OLAP only need a backup from time to time. Backup is not important compared to OLTP |
| **Design** | DB design is application oriented. Example: Database design changes with industry like Retail, Airline, Banking, etc. | DB design is subject oriented. Example: Database design changes with subjects like sales, marketing, purchasing, etc. |
| **User type** | It is used by Data critical users like clerk, DBA & Data Base professionals. | Used by Data knowledge users like workers, managers, and CEO. |
| **Purpose** | Designed for real time business operations. | Designed for analysis of business measures by category and attributes. |

# OLTP vs. OLAP



| | | |
|---|---|---|
| **Performance metric** | Transaction throughput is the performance metric | Query throughput is the performance metric. |
| **Number of users** | This kind of Database users allows thousands of users. | This kind of Database allows only hundreds of users. |
| **Productivity** | It helps to Increase user's self-service and productivity | Help to Increase productivity of the business analysts. |
| **Challenge** | Data Warehouses historically have been a development project which may prove costly to build. | An OLAP cube is not an open SQL server data warehouse. Therefore, technical knowledge and experience is essential to manage the OLAP server. |
| **Process** | It provides fast result for daily used data. | It ensures that response to the query is quicker consistently. |

# OLAP Benefits

- OLAP creates a single platform for all types of business analytical needs which includes planning, budgeting, forecasting, and analysis.
- The main benefit of OLAP is the consistency of information and calculations.
- Easily apply security restrictions on users and objects to comply with regulations and protect sensitive data.

# OLTP Benefits

- It administers daily transactions of an organization.
- OLTP widens the customer base of an organization by simplifying individual processes.

# Thank you

@mitu_skillologies

@mITuSkillologies

@mitu_group

@mitu-skillologies

@MITUSkillologies

kaggle

@mituskillologies

**Web Resources**
https://mitu.co.in
http://tusharkute.com

@mituskillologies

**contact@mitu.co.in**

**tushar@tusharkute.com**