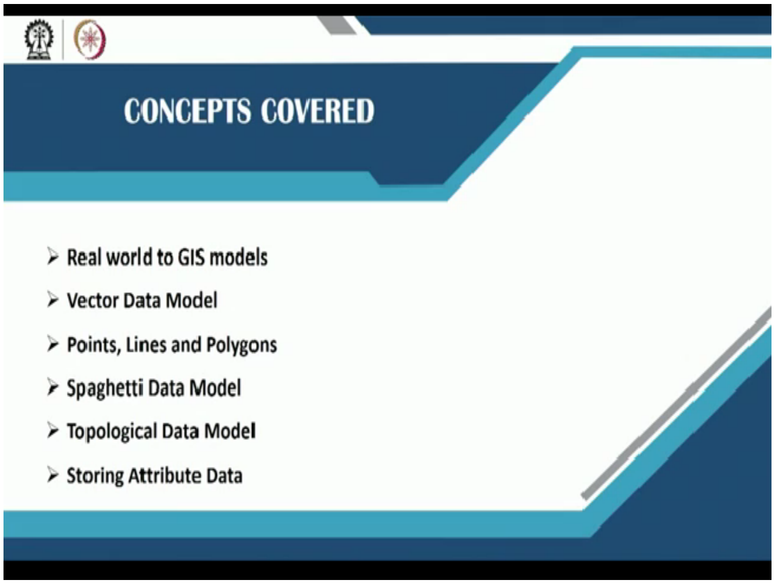
Geographic Information Systems
Prof. Bharath H Aithal
Ranbir and Chitra Gupta School of Infrastructure Design and Management
Indian Institute of Technology-Kharagpur
Module No. #03
Lecture No. #12
Representing the Real World (continued)

Hello namaste, I am back with the next set of lectures on data models wherein we will be looking at first lectures in terms of what do you mean by a vector data model.

(Refer Slide Time: 00:33)



And when we look at the previous class that we have understood what do we mean by a data model, a data model is in terms of how do we represent a real word representation using rules in to converting it to a data or a database, which actually can be variable it is a quantifiable or qualitative process and now in order to represent that we have different types of data, we have continuous data and we have a discrete data.

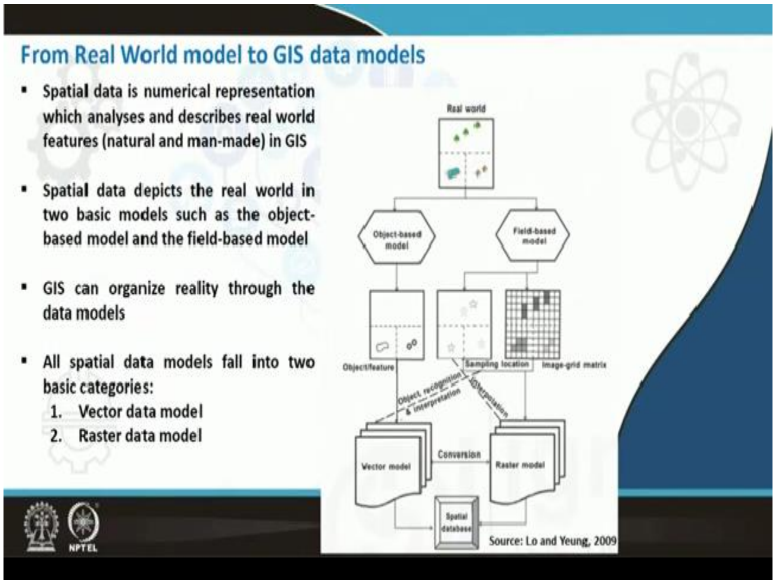
So when I say continuous data so you have a continuous set of values represented from a geographical location whereas you have a discrete data which has discrete values that is very specifically has a boundary and it is represented over a geographical location and this to represent the continuous or a discrete data we have fields and objects. Fields normally represent a continuous data and objects represents discrete data.

So when you are looking at this the first kind of data modern representation that you can find is nothing but a vector data model. In today's class you would understand what do you mean by a vector data model, how it is represented and what for example as I previously said vector data model is normally represented your point, line and the polygon, what are the different ways of storing a point, line and a polygon.

You have different data models, well known data models, a spaghetti data model and other one as a topological data model. So both of the data models can be used spaghetti as a very simple data model whereas topology is a quite complex data model, so we look at both of these data models and the last way is how do you store your attribute data, how do you store the entire data that you have and how do you retrieve it.

So what are the different methods that now you have to store and if the certain errors that should not be always performed in your database that also we will look at in this particular class.

(Refer Slide Time: 02:38)

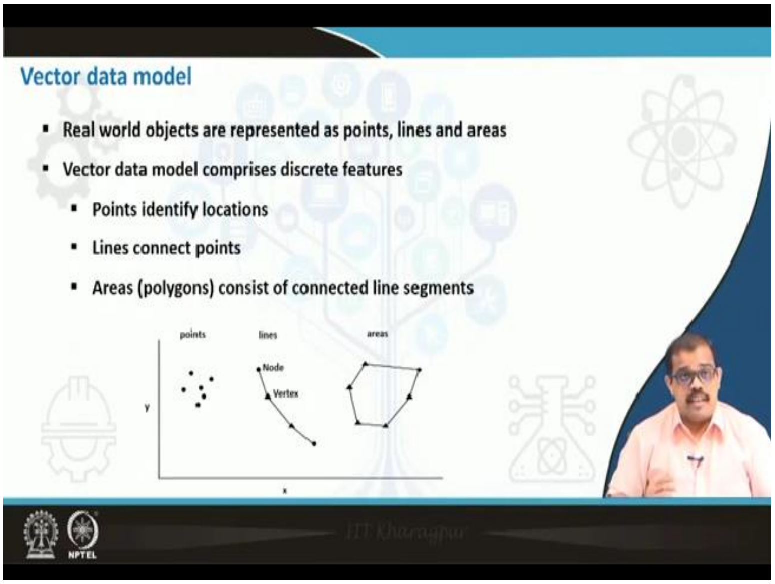


So now when we are looking at from the real word model to GIS data model spatial data as I said is a numeric representation which analyzes describes real world features in GIS, that is what is nothing but a spatial data. So we when I say spatial data you have a real world data with the geographic coordinate associated with, that is nothing but a spatial data. Now spatial data depicts a real world in basically 2 different data models.

One is called an object based model, other one is called a field based model okay, when I say object, object model is can easy has a different entities which are separable whereas field data model has more of a continuous data. So GIS can organize a reality through the data models okay, it can completely convert the real data to the digital data through the data models.

When you look at all spatial data models you can either put them into vector data model or into a raster data model or both. So both of them are possible in the end in the same data model but these are in different layers okay.

(Refer Slide Time: 03:56)



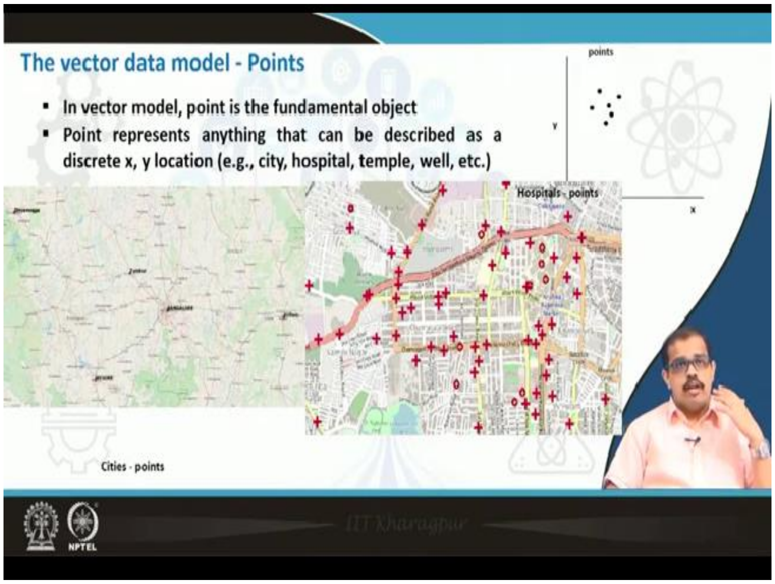
Now when we look at the vector data model each real-world objects are represented as a point, a line and a polygon. Let us take an example, we have let us assume that we have a building okay, we have a building and surrounding the building we have number of trees and next to the trees is the driveway or you have your roads that are actually there. So now if you have to represent this as a real world phenomena into your database.

The first thing is you capture building, the edges of a building that becomes a polygon because it is a closed loop of connecting line segments, then you have lines okay. So lines that are represented in a form for road. So when you are actually routing a road in the database you converted normally as a line, though it can be very clearly represented based on a polygon, but normally when you are actually representing in the real world in the data model you represent most of these roads as a connecting line segments.

Then you have points, for example it may be identifying the particular building or number of trees around the building. So all of these becomes a point, so number of points connected will form lines and each of the line segments that are connected forms areas, areas are nothing but polygons which are the representation of a vector data model. So whenever someone asks you what is a vector data model.

The first thing that you have to answer is it is a point, line and a polygon. So always this has to be remembered okay. So PLP is what you mean by a vector data model, the point, line and a polygon. So always every data on earth surface can be represented in a point, line and a polygon. So remember whenever you have a vector data model you are representing the real word features in points, lines and polygons, whichever data it is, whichever the real world phenomena it is okay.

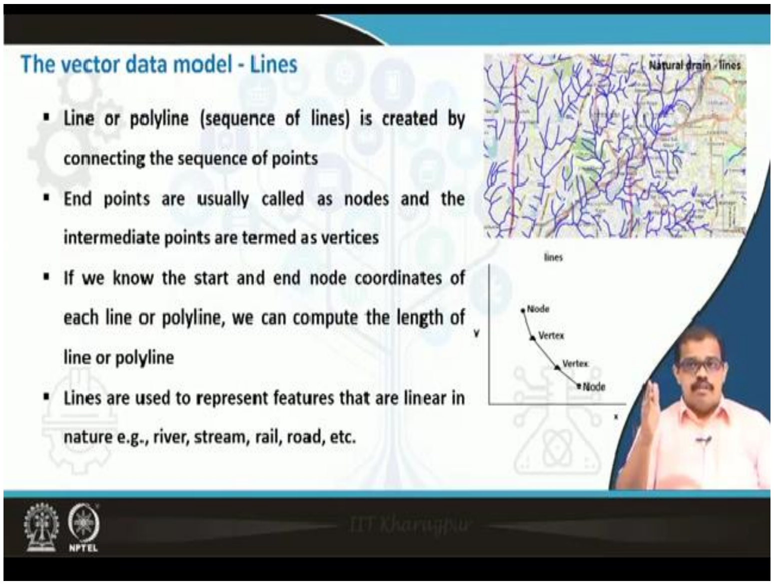
(Refer Slide Time: 06:06)



So when you are looking at a vector data model as points, points is a fundamental object, you can represent any building, any corner of a road but with a point, point represents a discrete please be careful here discrete x y location okay, it is represented only by a discrete x y location that is nothing but a points. A city may be a point but when you go into looking at city boundary it becomes a polygon okay.

But city, for example if you see there are number of cities that have been represented here, tier 1 and tier 2 cities okay, Bangalore is a point, Tumkur is a point, Mysore is a point, Vellore is a point okay, but if you are representing then the connection between Bangalore and Mysore then it becomes a line, number of line segment and if you are representing a boundary of Bangalore then it becomes a polygon okay.

(Refer Slide Time: 07:08)

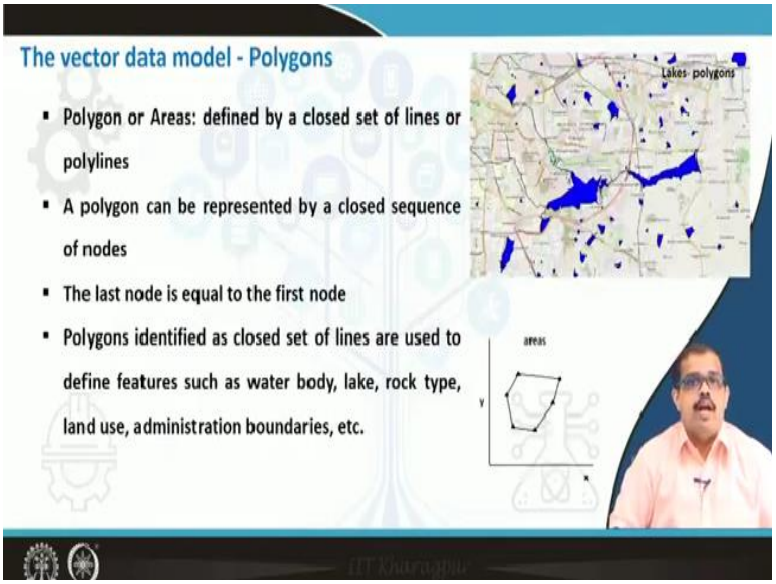


So now we understood what do you mean by points, point is just an x y coordinate okay, but when you have a line or a polyline is connected by sequence of points, if there are number of points that are connected as shown here, you have a node, you have a vertex, you have a vertex number of vertexes and finally you end with a node, this becomes a line okay. So now each end points are usually called as nodes.

Not then if you have one line segment here and then you have a connected then this becomes a vertex and not as a node, where it ends it becomes a node okay. So and the intermediate values are called as vertex or number of values are called vertices, if we know that the start and the node coordinates of each line or a polygon or a polyline we can compute the length of a polyline.

So which means length of a road, if you want to compute then you should know what is a starting point to nodes at the start node and an end node, then lines are used to represent features that are more linear in nature, for example it may be a river, it may be stream, rail, road all of these features. So yeah any features that has number of points that are connecting in a certain fashion from a start node to then no then it becomes a feature of a line.

(Refer Slide Time: 08:37)

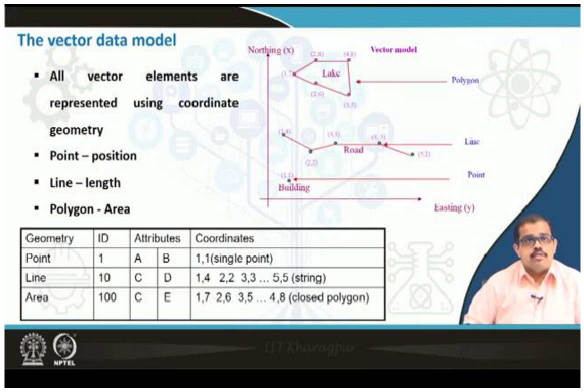


And when we look at the third kind of representation it is nothing but a polygons, if you see this map this is representing a number of lakes in the city of Bangalore okay. So when you see this is nothing but blue color is representing lakes, so when I say polygons or areas or surfaces this is defined by a closed set of lines or sometimes called as polylines okay. So you have if you see this image here now so you have this as a line okay second line third line.

So these are nodes okay not the vertices these are nodes node 1 and node 2, there is a start node and end node 4 this line, this is a start node and end node for this line. There is a start node and end node for this line, start node and end node for this line and start node and end node for this line finally reaching the same start node or the first line. So this becomes a completely a polygon okay.

So normally in a line the start node and end node are different whereas in a polygon the end node is nothing but a start node always okay. Polygons identified as close set of lines are used to define features such as water body, lake, rock type land use, administrative boundaries, all of these are the presented by a polygons. This example which I have given here is a set of lakes in south east of Bangalore where you can see these are number of lakes that have we have extracted from a map source okay.

(Refer Slide Time: 10:17)



So then when we look at the vector data model all vector elements are represented as using continuous geometry that has point represents a position, line, a length and the polygon and area. So when someone says find out the length then you have to look at the number of line segments that are there. If you are computing the same area so look at the entire loop of a polygon and look at the area, if you are looking at a point then it is only a point.

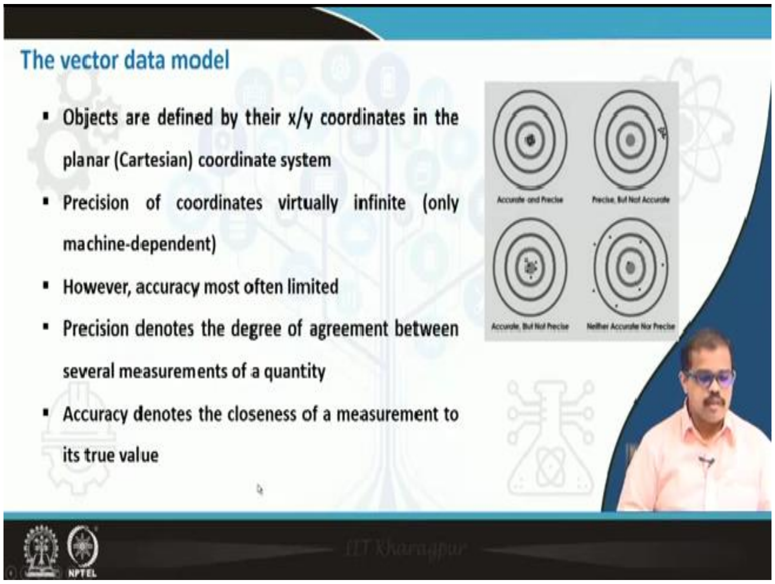
If for example if I have to give you the location of this particular building then I will just say the centroid point of this particular building which represent this particular buildings geographic location okay. So that is how in a vector data model it is represented point is nothing but a position, line is a length and polygon is an area, that is what I am representing it here.

If there is a lake here it is representing a point okay, when there is a line it is representing the connected line segments and is representing length and if it is a building a point here or even these points represent the position of that particular place, the geography coordinates okay. If you look at your database if you have a point line and a polygon given ideas 1, then attributes as a and b then the coordinates is 1 , 1 okay.

Because at points geographic location has a single coordinate, but when you look at line so if for example if I have given it as a unique id is 10 and attributes are c and d any I have just given a random attributes and when you look at its coordinates you will have a number of strings okay, start this is a starting node and this is an ending node okay. So when you have an area let us say that I have given the its unique id as 100 so it is a closed polygon.

It will have several number of coordinate values, so this is a starting node okay and this is the again another node but the end node will be the same as a starting node, it is a closed polygon okay. So this is how a vector data model is represented, I have not in the area part I have not represented the end node, so end node would be the same as a starting node, so please keep that in mind okay.

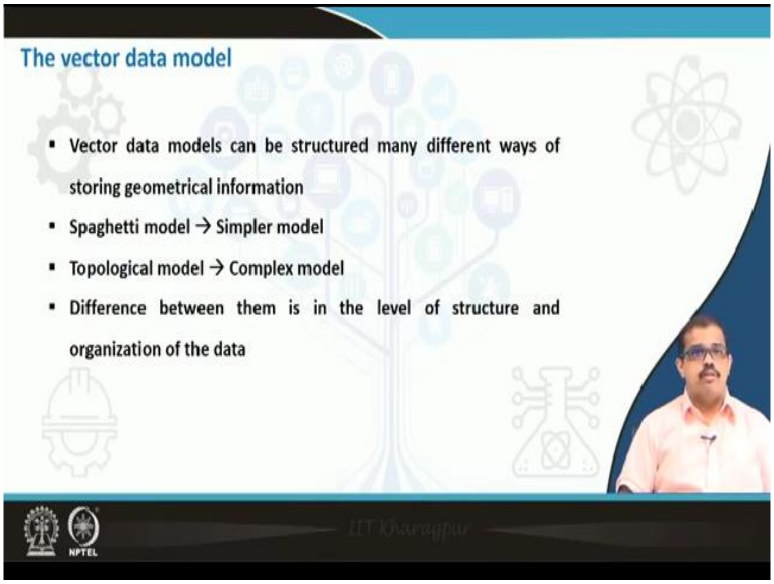
(Refer Slide Time: 12:52)



So in a vector data model as I said objects are defined by their x y coordinates in a planar coordinate form okay, when you look at the precision, the precision of coordinates virtually are infinite okay, but are quite machine dependent. So it is dependent on how you represent it how you actually work on it. However when you look at the accuracy most often is quite limited.

So this whatever when you look at the vector data model you have the accuracy which is quite limited but the precision denotes the degree of agreement between the several measurement of quantity. So it is not though we say it is not precise but the precision is dependent on what you are actually measuring on the ground or to the database. Then accuracies denotes the closeness of measurements of its true values but it is always quite limited in terms of understanding okay.

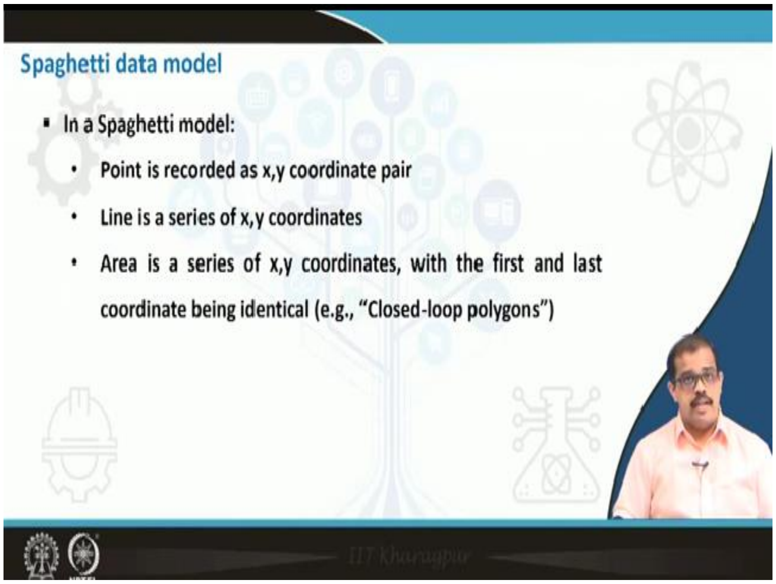
(Refer Slide Time: 13:56)



So when you look at the vector data models this can be structured and many different ways of storing geometrical information you have 2 types of vector data model, one is spaghetti model. Spaghetti model is quite simpler model it has very simple model in terms of representation whereas topological model is a quite complex model the way the data has to be represented is extremely complex.

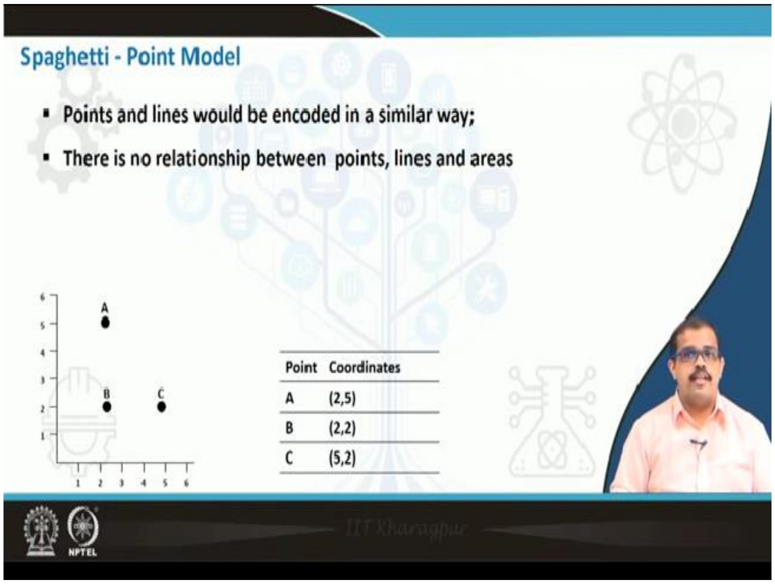
So difference between them is in the level of structure and organization of data, how the data is structured and how the data is organized. So that is a difference between these 2 types of models.

(Refer Slide Time: 14:33)



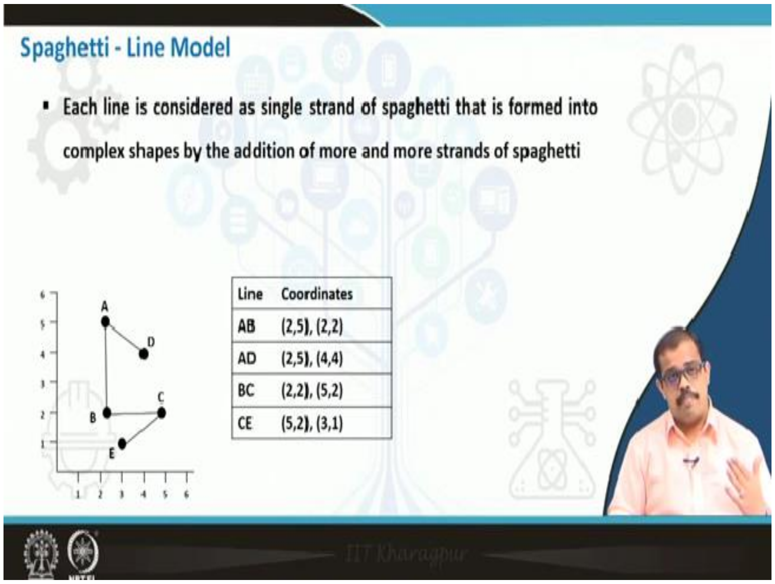
And when we look at the spaghetti model, spaghetti model the point is are represented in a form of x y coordinate pair like your vector data or in a form of a vector date. So line is a series of x y coordinates as represented the way. So we there is no change in definition area is a series of x y coordinates with the first and the last coordinate being identical, this is nothing but a closed loop it is the same definition.

(Refer Slide Time: 15:01)



I would not again deal into that but when you are looking at the point model points and lines would be encoded in a similar way, there is no difference, there is no relationship between point, lines and areas when you are looking at a spaghetti model. So if you have points here these do not have any relationship between points or lines and a polygon okay.

(Refer Slide Time: 15:27)

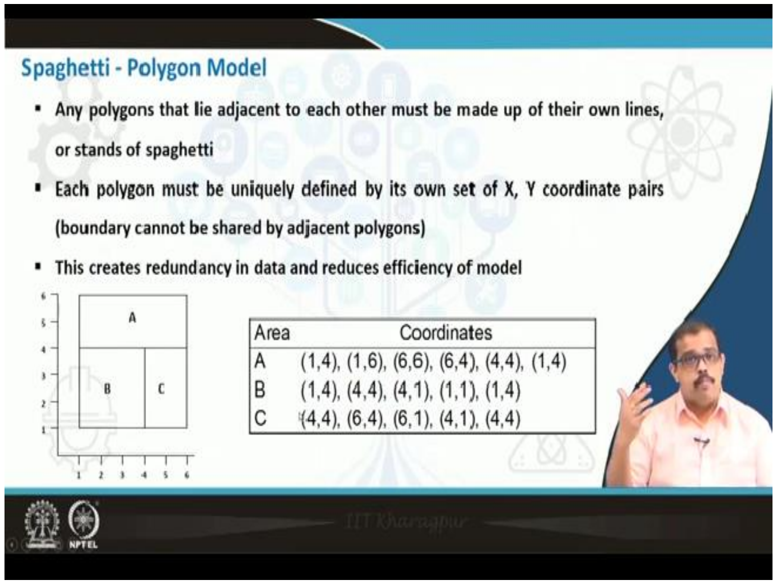


It is just the presentations and when you are looking at a line each line is consist a single strand okay. For example this is the entire line, this is your starting node is a, then you have vertices here, you have this line segment. So all of these are nothing but a single strand okay, each of this planned in this model. For example AD is a strand which I have represented here.

It has sir coordinates of 2 , 5 and 4 , 4, if you see A is 2 , 5 and D is 4 , 4, you see here it is 4 yeah it is 4 , 4 okay. So that is how you represent AD whereas when you look at AB that is 2 , 5 , 2 , 2 okay you see here the A is 2 representing 2 and 5 okay and when I whereas B is representing nothing but 2 , 2 here okay. So each of these segments is considered as a single strand, it is a complex shape.

But it is a single strand okay, they more and more strands of spaghetti can be added. These strands are called as nothing but a spaghetti. So you have points, points are represented, now you have line segments, each line segment is a stand which is represented here.

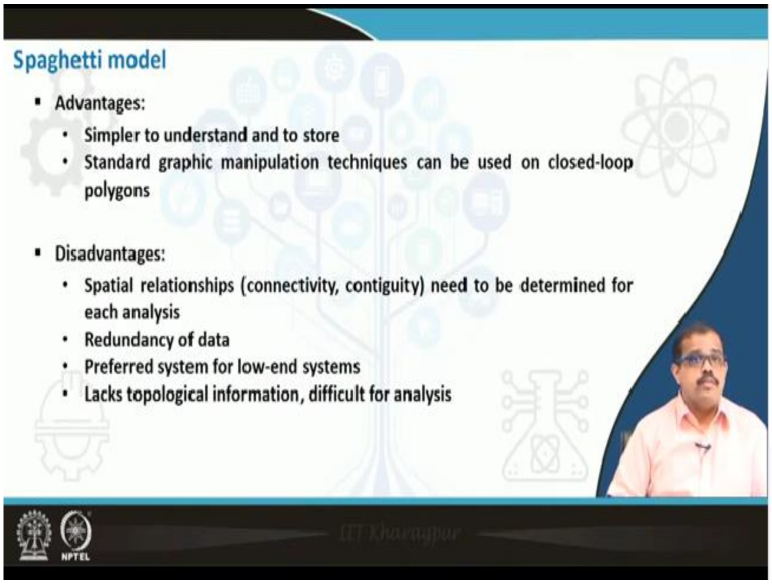
(Refer Slide Time: 16:58)



Now let us go into polygon model, so any polygon that lies adjacent to each other must be made up of at least 2 strands of spaghetti or 2 lines okay, each polygon must be uniquely defined by its own set of x y coordinate pairs okay, boundary cannot be shared by adjacent polygons, then that is one rule that it has okay, this creates if it shares it creates a redundancy in data. For example when I say A A is a polygon here okay.

It is bounded by 1 , 4 1 , 6 6 , 6 6 , 4 4 , 4 1 , 4 okay if you look at this this is 1 2 3 4 5 line segments 1 2 3 4 5 line segments okay. So 6 line segments, now when you look at this 1 2 3 4 line segment 1 , 4 is a starting point here, if you look at this this is one point 2 3 and just the line okay. So this is how the polygon model is built.

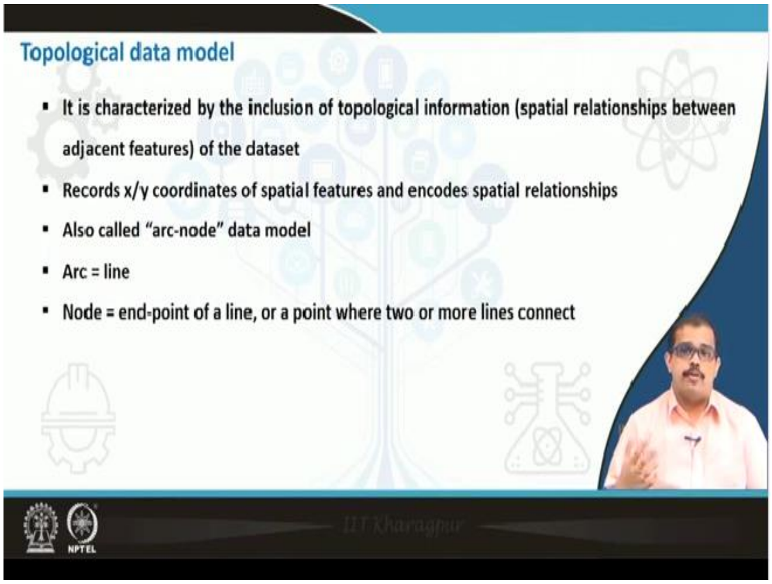
(Refer Slide Time: 18:15)



Now you when you look at advantages of this particular model it is simpler to understand and store you do not have to use much of the computational capability also, then you have standard graphic models which are standard graphic manipulation techniques that can be used on a closed loop polygons. The only the major disadvantages here is the spatial relationships need to be determined for each analysis, it cannot add the connectivity and continue otherwise it, cannot be used in its own way.

Then redundancy of data is there in the entire system the data is quite redundant, then you have preferred system for very low end system because you if you have a system where it is not computationally capable then spaghetti data model is something that you have to look at and very importantly is that the geographical information of the topological information is difficult to analyse in it or it lacks of topological information basically because everything is represented as point, line and polygons and the entire thing is nothing but strands and spaghetti.

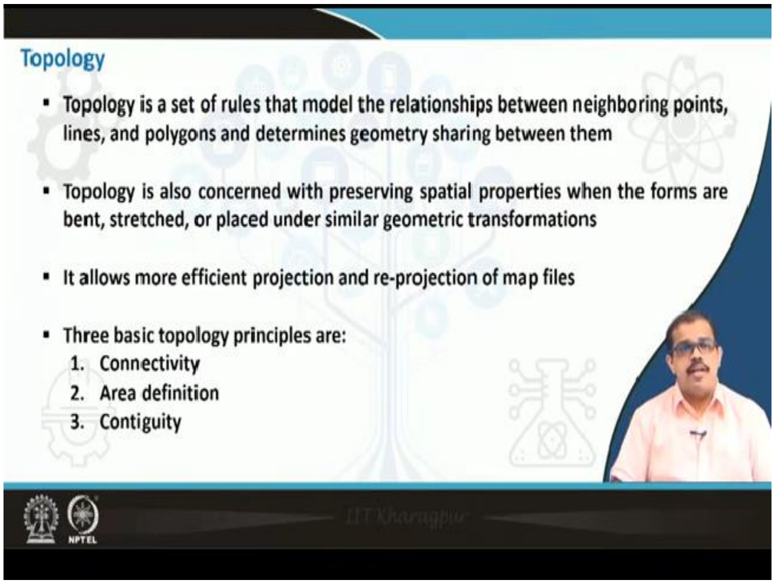
(Refer Slide Time: 19:29)



So but when we look at the topological model it is characterized by inclusion of topological information, whatever as a disadvantage there is an advantage here. The spatial relationship between adjacent features in a data set always exists, whereas when you look at this spaghetti model you always used to compute the relationships between them, whereas here you know what is the relationship which exists all throughout the way you are storing.

Then records x y coordinates of a special features and encodes special relationship between them it is also called as arc node data model because normally it is in the form of a line and a node. So arc is nothing but alone, line and node is nothing but the end point of a line or a point 2 or more line connects that is nothing but a node. So it is in the model is called arc line or arc node sorry arc node kind of a model. So topological model is also called as arc node model where it is arc is melting but the line know at this endpoint of a line.

(Refer Slide Time: 20:33)

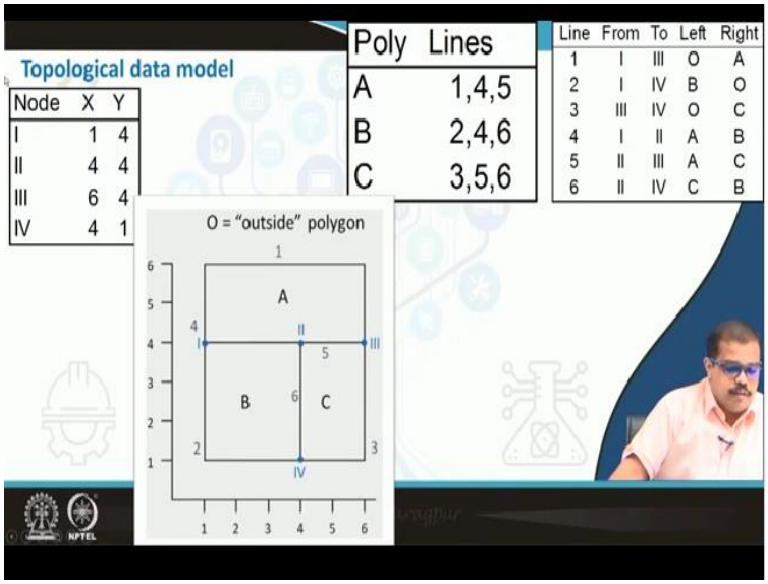


So when we are looking at the topology, topology is a set of rules okay be specific here, topology is a set of rules that model the relationship between the neighbouring points okay. So in a spaghetti model we are not worried about point, lines and a polygon. So it is about points, it is about lines, it is about polygons. So it is completely different, but here it is between neighbouring points, lines and the polygons okay.

It is a relationship between neighboring lines and points, lines and a polygon determines geometry sharing between them always okay. Topology is also concerned with preserving spatial properties when the forms are bent, stretched or placed under similar geometrical transformations which are not happening in the spaghetti mode. Three basic topological principles you have to always look at this connectivity.

How do you define connectivity, how do you define area, how do you define the continuity. If you understand all of these 3 things then topological data model representation is much extremely important or easier okay.

(Refer Slide Time: 21:47)

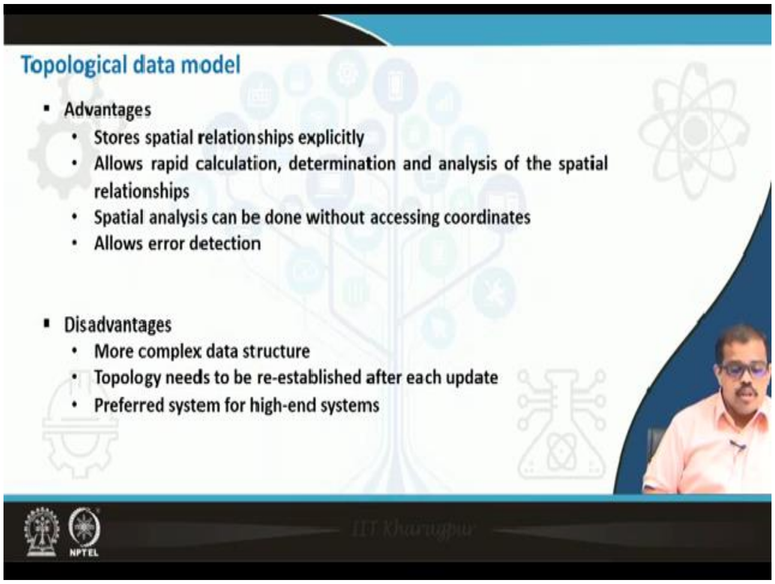


When we look at the topological data model, for example here if you are looking at this okay the first node 1 is a point that is represented between 1 and 4, if you look at this particular node 1 in draw if you these are connected by 1 and 4, whereas node 2 is about 4 and 4 this is connected by 4 okay, when you look at the if you see this it is 4 and 4 okay, when the node 3 is 6 and 4 okay.

And when you look at the node 4 it is between 4 here and 1, that is why you represent a point and when you are representing a polygon here polylines A, for example if we are representing any of those polylines it is represented in the form of 1 4 and 5, it this particular polygon is bounded by 1 4 and 5. All of these 3 are different polylines okay. So similarly we represent the polygons so each of this line based on the line from where to where okay in the left of that particular polygon you have an outside polygon.

You do not have any I mean when I say outside polygon it means to say that there is nothing in between them and on the right you have a polygon which is named on it. So that is how we read it, so topology this is for when you look at this it is a point this is when you look at it, it as line and when you look at this it is about a polygon. So point, line and a polygon. So always topological model has topological data inbuilt in it.

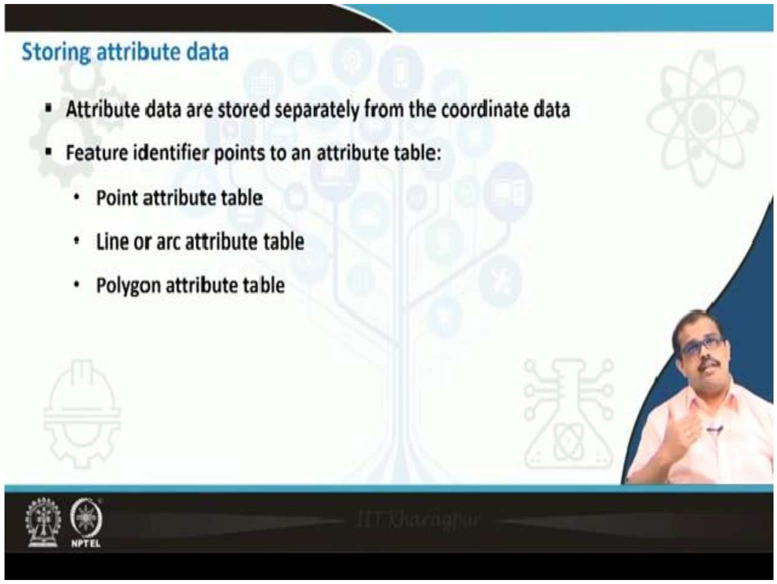
(Refer Slide Time: 23:42)



That is a biggest advantage of considering a topological data. So when you are looking at advantage its stores spatial relationship explicitly okay which is not done in a spaghetti model, it allows rapid calculation, determination, analysis of spatial relationships which is extremely important. Spatial analysis can be done without accessing coordinates you do not need coordinates.

But you can do spatial analysis and always allows error detection that is extremely important in terms of when you are representing a data model, more complex disadvantages it is more it has a very complex data structure, topology needs to be re-established after each update, preferred system for very high-end system, if you have a very high-end system and it is very highly preferred okay.

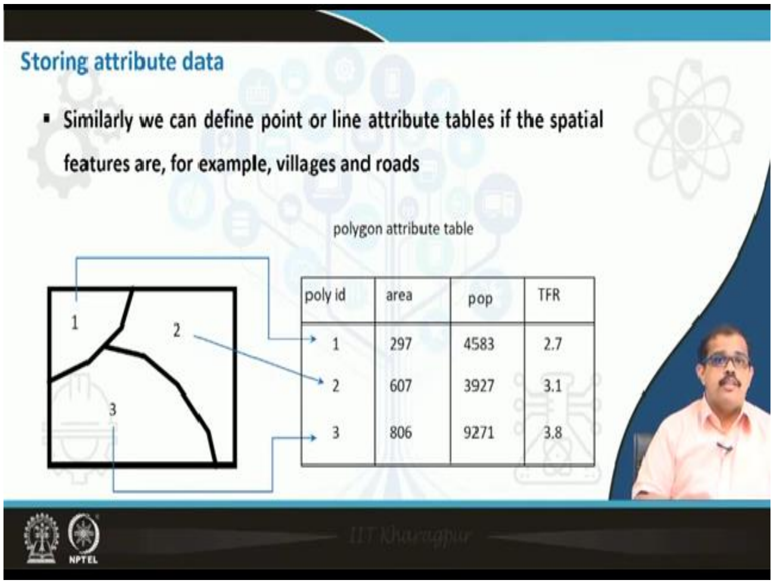
(Refer Slide Time: 24:35)



So the we are coming to the last part of this particular lecture, so we have understood what how are different of data models, the vector data models, the spaghetti and the topological data model. Now we look at how do we store an attribute data, so far when the attribute dates are normally stored separately from the coordinate data or in the same form but the coordinate information is in a different columns whereas attribute data is in a different column.

So when you are looking at feature identifier points or to an attribute table whenever you look at a feature it gives you an attribute table, point attribute table line or an arc attribute table and a polygon attribute table. So each of these would have a separate attribute table connected to it okay.

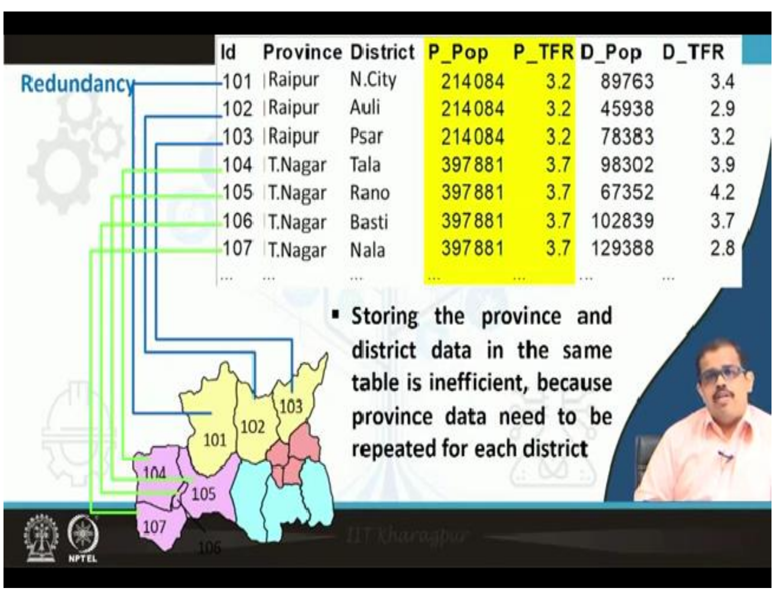
(Refer Slide Time: 25:22)



Similarly we can define points or line or attribute tables if the spatial features are for example if it is a village road etc., then you can define the attribute table in this way. For example let us say this is a polygon here okay, this is another polygon right, this is the third polygon okay I am representing area of this particular polygons here, the population here and the TFR. So this gives there is nothing but my attribute information okay.

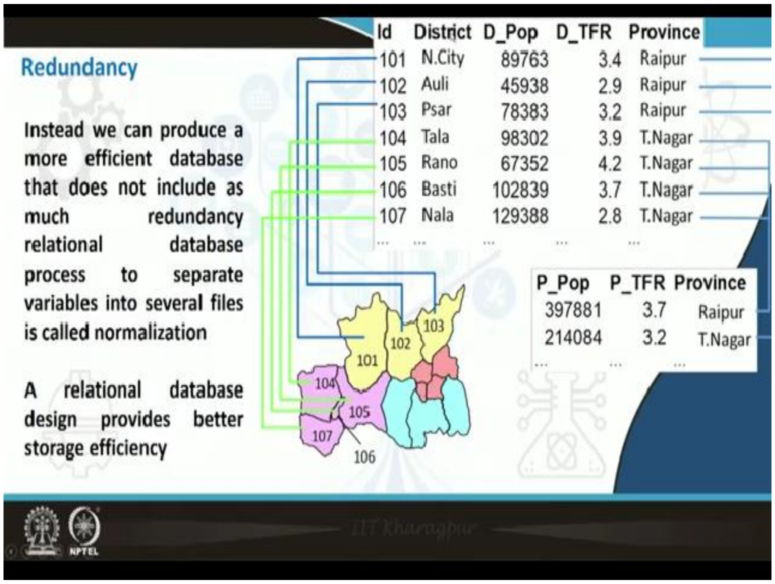
So the same id will have the x y location of this particular polygon where it this x y, this x y, this x y, this x y and finally this and this. So that will give me the entire information about that polygon. So you have geographical information also, you have attribute information also.

(Refer Slide Time: 26:16)



The very important point of converting into data base is to avoid redundancy. So when you are storing for example the this data, this about the province and the district data in the same table is quite inefficient, why because province data need to be repeated for each district right. So if you have number of district for example here if you look at you have the same province that is repeated for all of these districts. So this becomes redundant, the population becomes redundant.

(Refer Slide Time: 26:56)



But if you are represent something like this, so you have unique idea is represented here, you have a district that is represented here, you have a population, then you have a province. Now this becomes more efficient database in terms of if you are trying to put it as a relational database. So you will say that these each and every id has a particular relation that is attributed to it. So may you build a relational database and you represent the province like this.

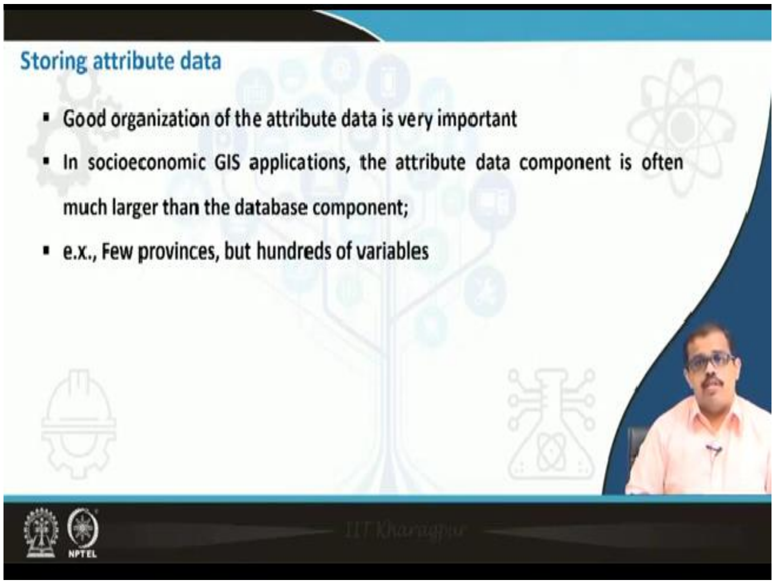
So that becomes a relational database which is actually relating all your attribute data to your id. So redundancy is the one that you have to remove from your database. Now if you ask me how do I actually remove the entire redundancy. Now what I have done basically here is, I have given province in a different database. If you have seen here Raipur and T. Nagar is represented here instead of representing it in the main database I put it here.

So when I connect this with relations it becomes the same database, so you only thing is that it is stored separately, so that it is not in the same database you do not repeat it in the same database.

So now if you ask me why and how you actually have different mean how do you remove this redundancy then it is how you do a normalization. So when we look at different databases, different ways of representing a database I will speak about normalization of a database .

How different normalization you have first normalization, second step normalization, so we look at a first form, second form, third form etc. when we are looking at the database but as of now it can be done using normalization okay. So normally a relational database design provides the best storage efficiency and tell you what do you mean by relational database. Now you have object-oriented databases etc. So we look at all of these databases and when we come to the database point but as of now we have to remove redundancy so you need to have a normalization for your data okay.

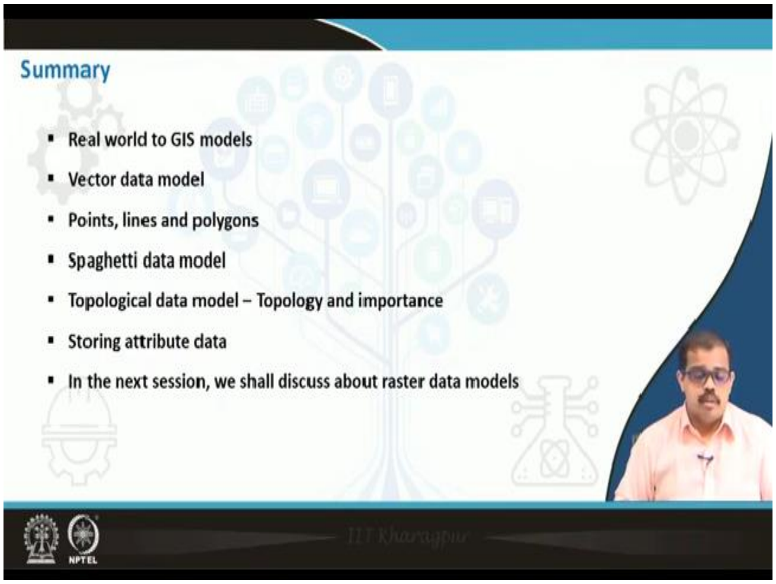
(Refer Slide Time: 29:14)



So that is what you should remember then when you are storing attribute data at any whenever you have any data it is attribute data that has to be stored. So always just storing a geographic coordinate will not give you any output, when you have attribute information added to it then it gives added value to that database okay and so for example in socioeconomic GIS applications I attribute data component is often a very large component in a database.

If you do not have an attribute data then that particular thing is not even called a database, it can be just a geographic base okay. For example few provinces 100 of variables, so huge of source coming any a socio-economic surveys any physical survey etc. will have a huge amount of attribute data without attribute data your database has no value okay.

(Refer Slide Time: 30:11)



So that that is what I wanted to speak about the subject in this class, we have looked at what is a vector data model, it is a point, line and a polygon basically a spaghetti we have data model we have looked at. So each point is represented as its own and each of the line has different strands that is represented in a form of a coordinates then you have polygons.

Each of the polygons are represented by different strands that are connected together each of the strands together I am a starting node to the end node forms nothing but a entire data model of a spaghetti, when you look at topological data model it is nothing but the arc node datum on arc is nothing but a line, node it is nothing but a point that is connecting each of these line segments or is the end start an endpoint in a polygon model.

Then you have storing attributes data. How do you store an attribute data, I spoke about redundancy, you can always remove redundant data and make it another database whenever it is required it just populates into your sheet wherever you want to do the analysis. Then I also spoke about different attributes, how do you actually look at attributes data, how do you look at different data bases, how do you actually move redundancy.

The main form of removing redundancy is using normalization form you have different normalization form which we will discuss. So as of now this is what we had to discuss for today. In the next session we would look at different raster data models. So I would only speak about one kind of raster data models, so that is called a squat tree data model. So the rest probably I will leave it out for you to explore okay, thank you very much, let us meet in the next class.