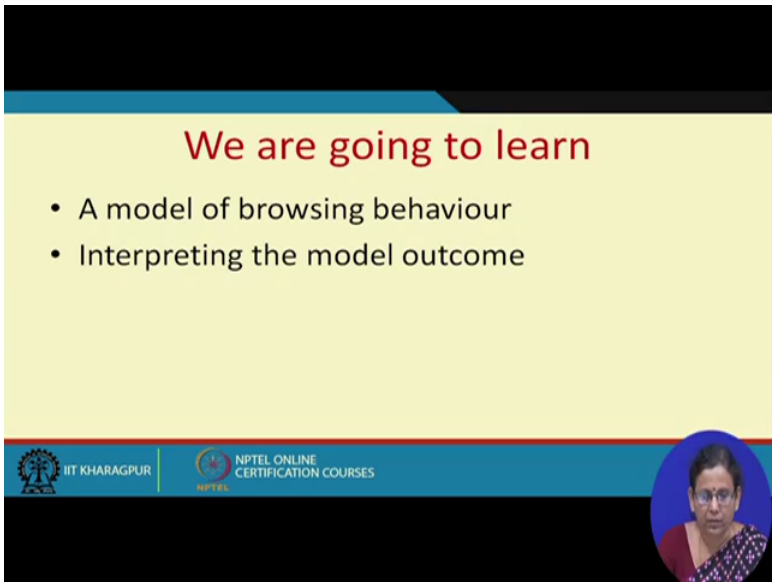
**E-Business.**
**Professor Mamata Jenamani.**
**Department of Industrial and Systems Engineering.**
**Indian Institute of Technology, Kharagpur.**
**Lecture-48.**
**User Behaviour Modelling From Web Log.**

Welcome back, we continue our discussion on how to use weblog data. In fact, last few classes we have discussed how, what is the source of this weblog data and how it gets generated because of HTTP request and response, then we have seen how exactly to clean this log, because this log is, this log is erroneous because of many reasons and we saw that how exactly to clean this log and come up with with a flatfile which is usable for various purposes. Now assuming that such a flatfile exists, today's lecture we are going to learn how we can use that to study the user behaviour in the website, user's navigational behaviour, a particular user's navigational behaviour in the website.

(Refer Slide Time: 1:27)

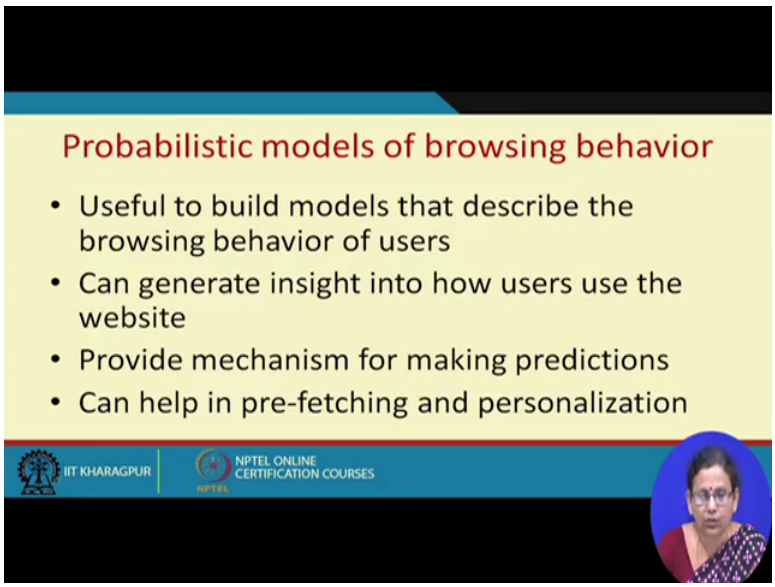



So in this lecture we are going to cease how to model the browsing behaviour and how to interpret this model outcome. In fact let me tell you in this series of lectures whether talking about various decision support situations to help the managers to take appropriate decisions in e-business scenarios, the model that we will be covering are not, does not, I mean the whole list of the model that is available for e-business decision-making we are not going to discuss.

We will be looking at the time are available to us, we will be limiting ourselves to only few models. So you should not be getting the idea that these are the only decision support models available in e-business setting. In fact as you know, this e-business is about automating business processes using informational communication technology. And I have made it clear from earlier lectures that even the traditional decision-making situations where you use informational communication technology, they also use this e-business because a business does not change the meaning of business, it is simply improving the business process using technology.

So the model that we are going to study right now is one such model which one can use to study the user behaviour in a website and take appropriate decisions. These decisions can be, maybe to improve the navigation structure of the website or it may be for deciding how the contents should be managed and so on.

(Refer Slide Time: 3:38)




The model, look, we are talking, we are going to talk about the probabilistic model of user user's browsing behaviour. Now user's activity in a website is pretty random, he can come to, he can directly come to your website by typing your web address if he knows, while searching, while he performed searching operation in some search engine, if as a result of search your website, any page of your website appears, then he can enter, he can browse through your website, the amount of time he spends on each page is also varies. So this behaviour of this, browsing behaviour of this is completely random.

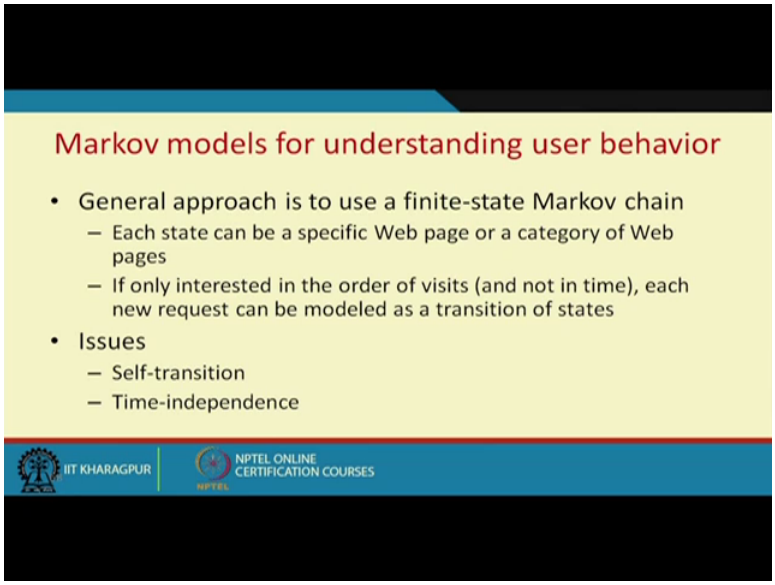
So therefore we will be using a probabilistic model to study the browsing behaviour. Okay. These transitions that users will be making from one place to the other can be modelled as Markov process. If some of you know about the Markov process, then it is fine, but anyway I am not going to, going deeper into the mathematical details but that should not again give you the idea that the way I am talking in a much higher level is how such models are implemented. In fact all the decision support models, though will be giving just overall view but for actual implementation, you need to know, you need to have many expertise.

You need to have programming expertise of course but you also have to have the modelling experience. So mathematics is the key but I will be making, in all my lectures I will be making the situation very simplistic so that even without a thorough mathematical background you can understand it and I expect that later on you will be studying more on those mathematics and can pick up. Okay. So now in such situations where the behaviour is random and the user is going from one place to the other we are trying to model it as a Markov process.

Now Markov process is of many type, it can be finite state, it can be infinite state, it can be discrete time, it can be continuous time and so on. But here specifically we will be using a discrete, infinite state and finite state Markov process and we are going to see you like how parameters of that process can be obtained. Now here in a typical Markov process, you have a number of states. And the, by finite state we mean the number of states are finite further but there are Markov processes where the states are infinite as well. But here we have modelled it as a finite state Markov process. Okay. So now if you are going to model it as a finite state Markov process, then what are the steps, the steps here are the webpages.

Now because the transition will be made from one state to the other, so from one page the user will be going to the other page. So when it goes, it makes its transition. Now the number of, if we assume a state to be a page, earlier we read because that is the place from which the user is making a transition to another place. However, considering a very large website consisting of thousands of pages, in a typical corporate website will be containing thousands of pages, under this situation, considering each page as a state and building a model with thousands of states, you can of course build but building such a huge model is difficult to manage.

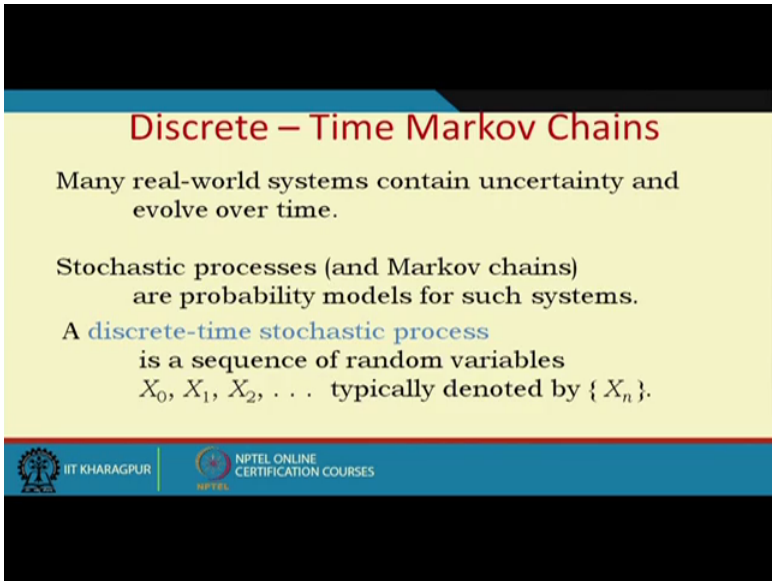
(Refer Slide Time: 8:33)




So therefore we further assume that we can club many webpages together to make one state. So each state can be a specific webpage or a category of webpages. Then if we are only interested in the order of the visit and not the time the user spends on each page, of course we can ignore the time part, but if at all we consider it, then we again have to think about how to integrate time into this model. So now each new request can be modelled as a transition from one state to the other.

Here various issues are there, like self transition, somebody would like to make a transition from one page to itself, maybe by refreshing also. Then time independence is something which is a part of, which is a specific feature of this finite state Markov process but we are going to include time here. Now how you are going to do it, let us see, let us just have a little bit idea about the discrete time Markov process.

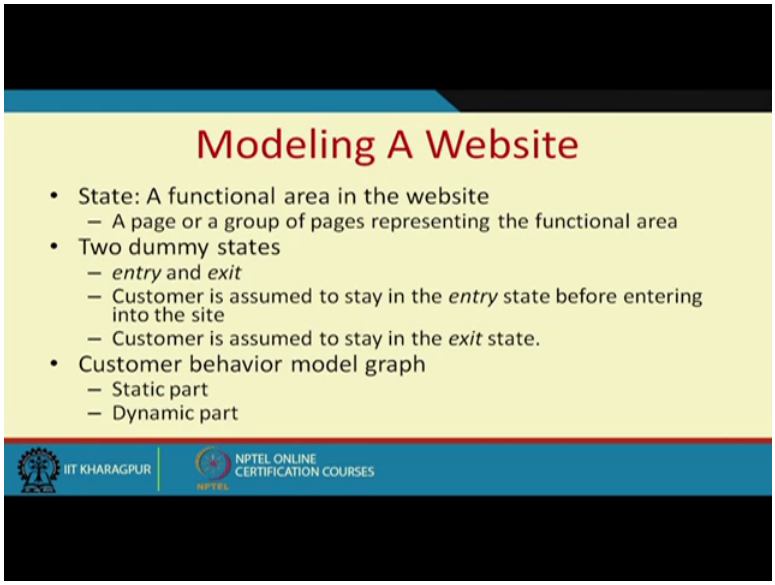
(Refer Slide Time: 10:17)




In fact from in the series of lectures on this decision support activities I have made it clear from the beginning that simply knowing about, that here there will be optimisation, there will be simulation model, really does not help much, unless otherwise you actually use it. So therefore we will little bit more technical, the context will be little bit more technical to incorporate the feel of how exactly such decision support models work. So many real-world systems containing uncertainty which evolved over the time can be modelled as discrete time Markov process.

So this is a stochastic process and this is not the only stochastic process, there are many more. Typically a discrete time stochastic process is a sequence of random variables, X0, X1, X2, etc. and these random variables are nothing but the states, one state coming the other, 1st, how do we define our state, we define our state of the web page or a group of webpages which we can think of as a specific category. Now from category let us say about the company, somebody is going to another category, say product search, from product search he is going to another category, say making payment decisions and so on.

(Refer Slide Time: 12:23)




So here is the Xs are the states. Now if you put, try to put, try to model, in this line if you try to model our website, state we define as the functional area in the website, it is a page or a group of pages representing the functional area. Now to the website the user can enter in many ways, he can come from another page to a specific page you can come to, come to that page from another page which is within the site, he may be coming to that which from the webpage of some other webpage which does not belong to your company but your company has been referred there.

He can come to your page through, while searching through, searching for some product or service through the website and through the search engine, so therefore we assume here, because you are building a model after all, so in this model we assume that before the person, before the user enters into your website, he stays somewhere in someplace before coming into your site. That can be through a search engine, that can be webpages or web page of some other organisation or he may be staying in some state before writing your web addresses.

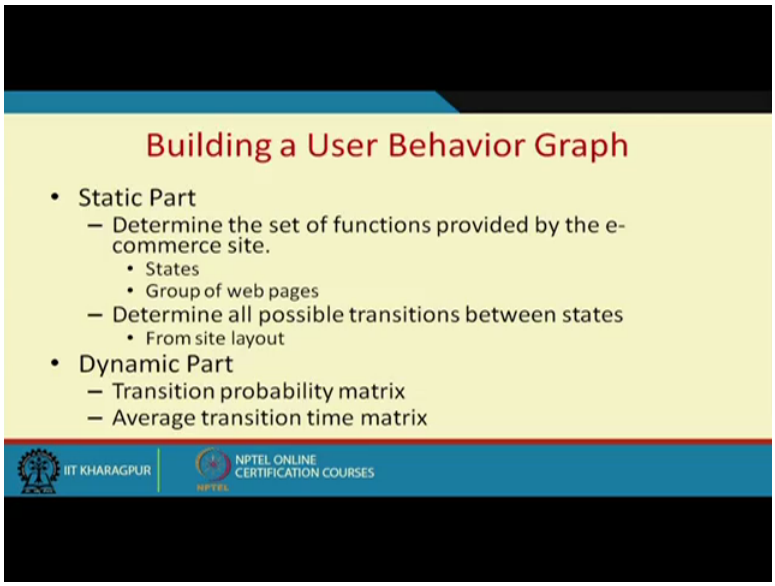
We can directly write this but whatever may be the case, we assume that before coming to the site the user stays in the entry state. Okay. Then, when his session is over, see what is a session, we defined last class, last feud lectures we have been talking about how to extend the sessions and all and again while talking about the web log, we were talking about if we will called referrer, that referrer field actually tell from which previous page you are coming. Okay, so if that referrer page tells that you have come from either a search engine or by

directly typing the web address or from some other organisation's webpage, then it is assumed that you were in the entry state. Okay.

Then after the session is over, after the browsing is over, the user will be going to go out of your website. Then where does your user go, you do not, you cannot have any, he basically he does not matter, he does not return, within that session, then we can say he is going to the exit status of. So this entry and exit state we assume to be 2 dummy states. So the customer is assumed to stay in the entry state before entering into the site and customer is assumed to stay in the exit state after he goes out of the site.

So for modelling convenience we have assumed these 2 dummy states. Now assuming these 2 dummy states and considering the other states in the website, now what is the state in the website, either it can be a page or a group of pages with a with some common theme, okay. You can say it is a category, specific category of webpages. So, the graph that will be building, it is basically state transition behaviour of the user, we call this graph as user customer behaviour model graph. This customer behaviour model graph is going to have 2 parts, the static part and dynamic parts.

(Refer Slide Time: 17:30)

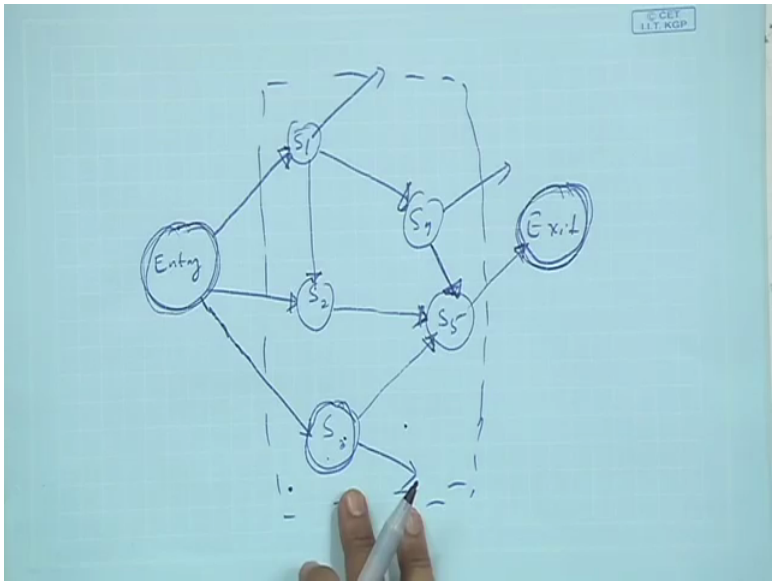



The static part, this state transition to presentation will be there, in the line apart, the time component will be there. Now, while building this user behaviour model graph which is nothing but a Markov process, in order to build the static part, 1st of all you come up with your states and add 2 dummy states, so 1st you make your state space. Then, to determine all

the possible transitions between the States. How do you do it? If you have knowledge of, this requires the domain knowledge about your website.

How the webpages link because a website, in a website the, from one page there will be links to other pages through form of menus, through form of sidebars, etc. But they are supposed to be connected. So looking at that connectivity detail, you 1st make a connected view of your website. So in that you will have the states as webpages and the link between 2 states as the edge between the website.

(Refer Slide Time: 18:27)



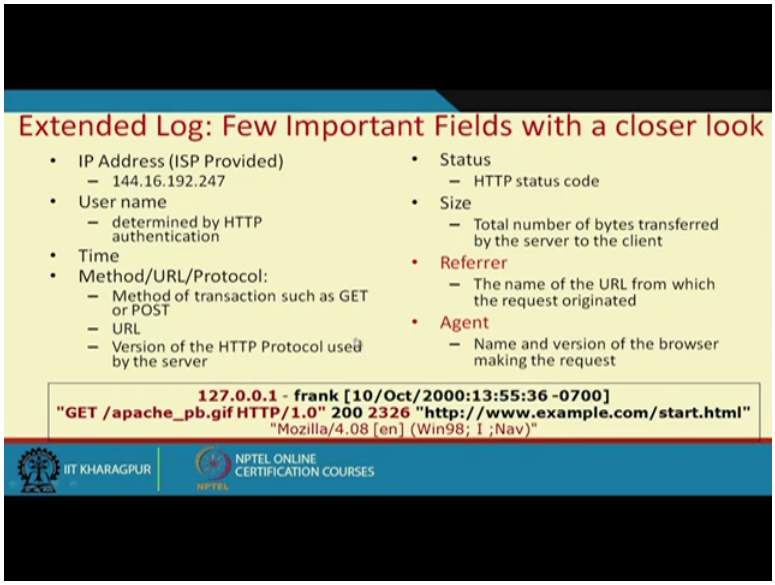

So the site will have a static part and a dynamic part while the model it. So 1st, as per our assumptions, we have made 2 dummy states, one entry state and 1 exit state. We assume that whenever the users come into the site, before they come into the site all of them stay in the entry state and after they go out of the site, they will be there at the exit state. In fact, it may, while you build the static part, you have the knowledge of how these states which are either webpages or a group of webpages, how they are connected with each other.

So this connectivity you can get from a site map or you can, or you can make a crawler for your website who will be visiting from one link to the other and tell you how the pages are connected with each other. Once this part is made, then from entry you can, from outside you can enter to the website through, to any of the pages, in fact here I connected only 3 pages but you can enter into any of the pages, similarly the user can exit the site from any page he likes.

So this, while building this static part, you simply you have to have the knowledge of how the webpages are connected, then we look at the data that is available to us from the access log,

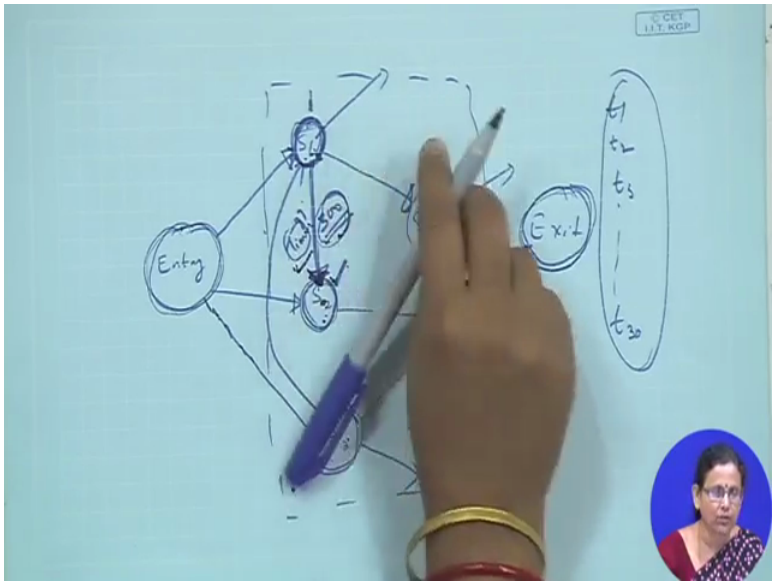
in fact the cleaned access log, after the preprocessed, whatever the state the access log is from, that is our input to make the dynamic part of it. Look, from this access log details, now we have now we know, you know, you have 2 fields, one, if we actually go back, I will just would like to show you that how various pages will be interlinked, that information, how will you get it.

(Refer Slide Time: 21:41)




Look, this is a simplistic form, this was your weblog, and you had a field called referrer. This referrer field tells you from which page you have come to which other page. Like let us say if your entry is for this page S2, then S2, where you have entered into S2 from S1, S1 is your, S1 will be there in your reference field. So looking that from which reference you have entered to which other page, you will know that how many people have made this transition happen. Let us say some, in a website, some 300 entries you found from your access log where the transitions are made from S1 to S2, where the referrer field was S1 and the page was S2.

(Refer Slide Time: 23:34)



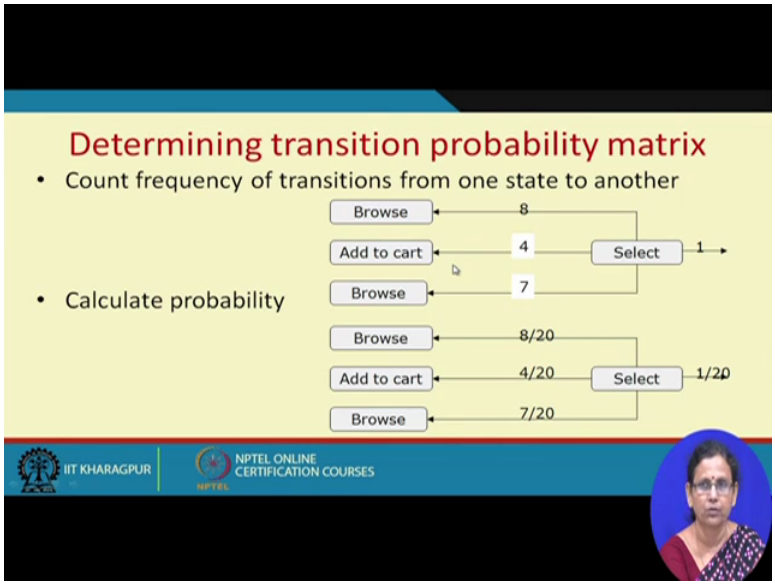

The page here is named here, this is a place for the page and what is the referrer, referrer was this one. So from this page you have come to this page. Okay. Then here, there is another field called the time, this time indicates the time at which this page was dispatched. Now if you go back, suppose the referrer found this to be S1, then a visitor in S1, S1 was also rendered by the website, so there will be some details of the S1 where maybe possibly user has come to S1 from some other state, let us say from S3. So in that case, that entry in the access log will also have the timestamp.

The timestamp of this one - the timestamp of this one, the time difference between these 2 is the time associated with this link. So there will be some time associated with this link. So this time again, for one user this time will be let us say T1, for 2nd user this time will be T2, for 3rd user this time will be T3, so there are 300 transition is made from S1 to S2, so there will be some T300, so these many number of time differences you are going to get. So all these times together, if you take the average, that is what for modelling proposal going to see, going to assume.

You can in fact assume this to be a distribution as well but we are not going to make the situation that complicated and we are saying this is just an average time. So the total 300 number of transitions and the average time. So with link these 2 parameters you can find out from your access log. So, now we continue our discussion on modelling the website and we understood that it has a static part and dynamic part and about how to construct the dynamic, how to construct the static part, I told you, you can use your site map if it is available, otherwise you can simply write crawler who will be, which will be visiting from one link

together and tell you how the pages are interconnected. But that is your own way you can determine.

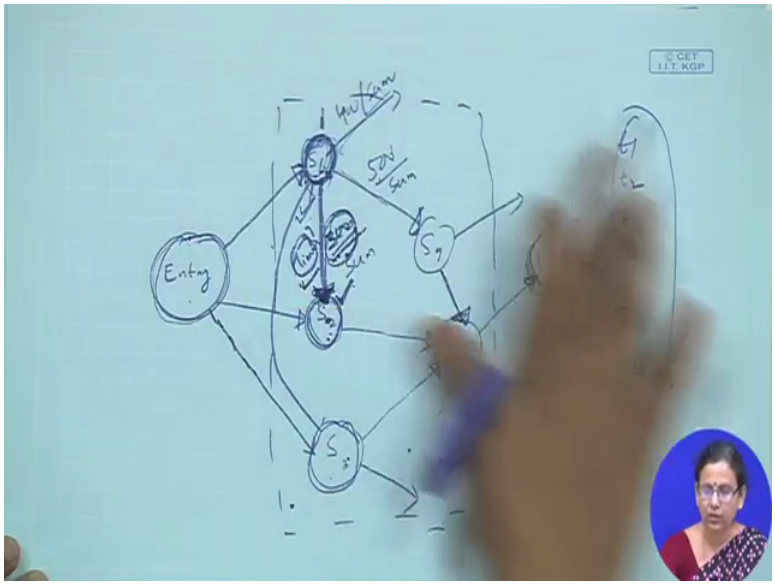
(Refer Slide Time: 25:55)




Now this is what I was trying to tell you we have, we are going from select State block state, one can go from select State to add to cart state, from select to payment state. And everywhere the number of transitions made, how do you get the number of transitions, you are going to get this number of transitions from your access log data where you have already cleaned it and you have discovered the time associated with each of the state. And difference between these, the number of times or you have, you know the referrer, not the timestamp, you know the reference.

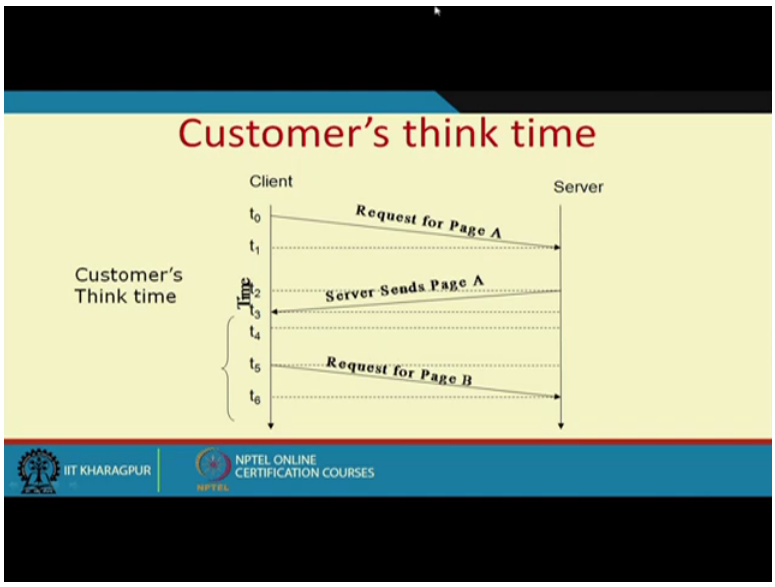
So therefore looking at the referrer, for this state the referrer is select, for this state referrer is select, for this state the referrer is also select. So from these details you know this. Now in a Markov process you need to have the transition probability values. So the total number of transitions made out of the select State is 8 + 4 + 7 + 1, this transition is leaving the site from here. So this 20, total 20 transitions have been made out of the select state so the transition probability associated with each of this link, now you can calculate, it is 8 by 20, 4 by 20, 7 by 20 and 1 by 20.

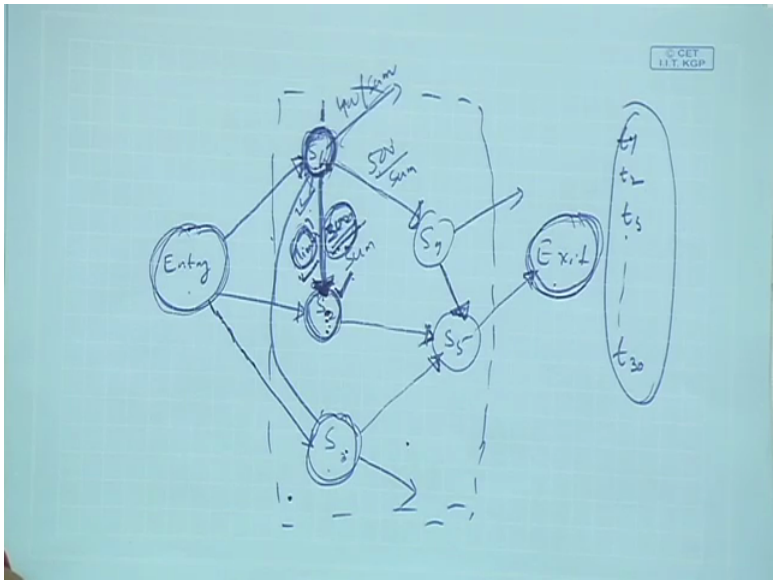
(Refer Slide Time: 28:12)




So this is how you calculate the transition probability values. So in the same manner for a bigger graph like this, where you have all the states and entry and exit states, looking at how many times the transition has been made and the average time, average time so far we have not discussed, looking at how many transitions made out from one state to any other state and the total number of transitions going out of a specific state, real estate is 400, here it is 500, so sum of all this, if we divide under each sum, you get the transitions transition probability. So this is how you can make a transition probability matrix for the entire site.

(Refer Slide Time: 28:41)





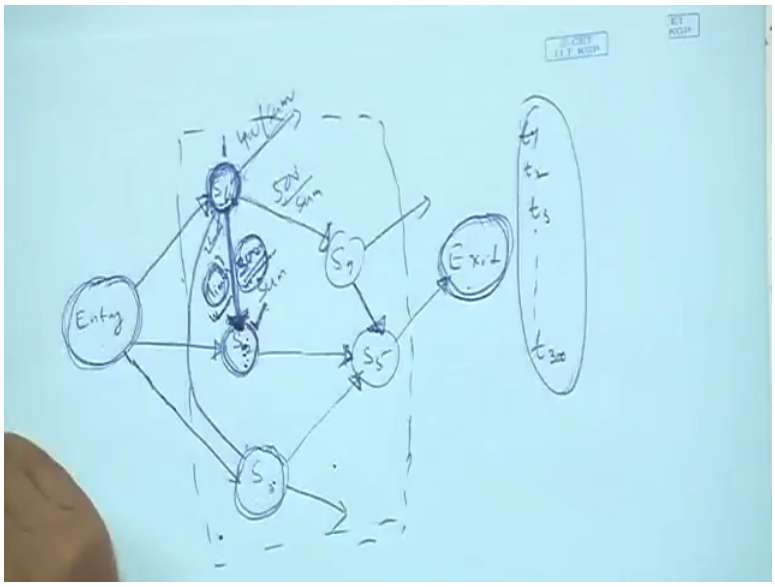

Now look at the time component, if you look at the time component, this is again your HTTP request response model and this particular figure you must have seen already. So user makes a request for page A, page, the server after getting the request, so there is some time spent here, then after getting the request the server takes some time to construct to the page, then it search the page, that again consumes some network time. So, the timestamp of the referrer field, I mean the referrer page whatsoever it is will be this one and when it was rendered. And similarly the timestamp for the other one will be when it was referred, referred here, also it is not depicted in this figure.

So the difference between these 2 times, difference between these 2 times, we call as the customers think time before he makes another transition. So this customer think time that I was talking about, from here to here, this actually includes many components. This is not the exact time when the user made a transition from here to here, this also includes the network time, the delays that involved in the in moving, sending the packets from the, from 1, from the host to the client and from client back to the host. + it includes the time for page construction at the server side, in case it is a dynamic page, the server has to construct it, getting the, fetching the data from the database or connecting to the ad server, it takes time, server takes some time.

So that time, + at the client's side, the browser against takes some time for collecting the packets and putting it together. So all these times together is the only thing which you can get. See, even if you really want that , you need a more finer detail removing this network delay, etc., it is not possible. So therefore let us be happy with this time, this particular time that we are getting. But the main important component is here is the difference between this

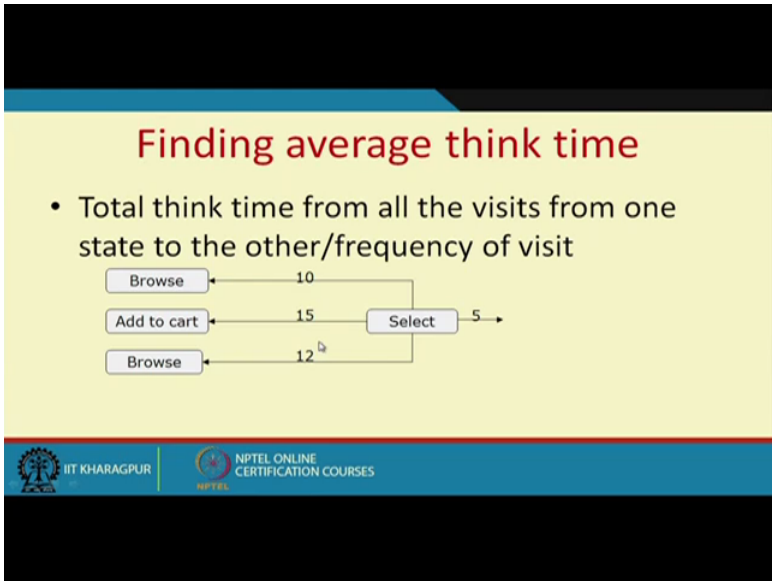
T3 and T5, this is the amount of time the user actually takes for deciding whether to visit, whether to send a request for page view or not, he may leave your side or he may send a request to another page.

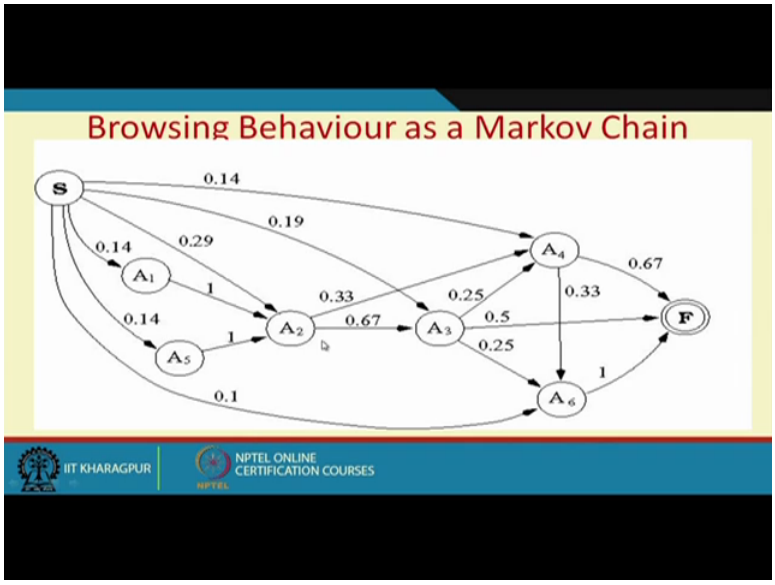
(Refer Slide Time: 32:17)



So the time also includes, this time difference that we are talking about also includes not only this network delay time, page rendering and reconstruction time, etc., this also includes this particular time where the user decides whether to visit another page or not. So whatsoever, this, all these times are now captured from the access log and these access log times, now we average it for over all the 300 visits that has been made from here to here and we can take its average.

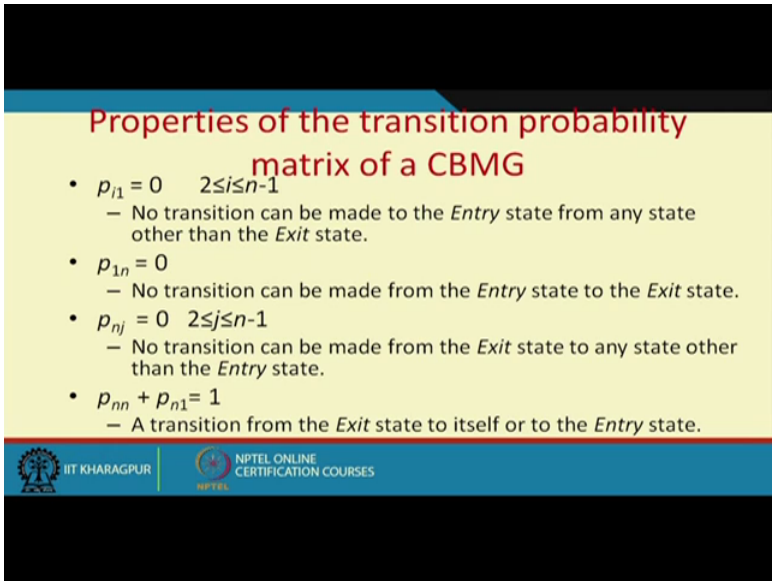
(Refer Slide Time: 32:29)






Okay, so you can take the total think time from all the visits, from one state to the other and you can divide it by the frequency of visit and you can reconstruct. So if you make the user behaviour as a Markov process, from your stat, you have a starting state, you have some exit state, some entry, this is your calling as entry state, this is we are calling as exit state and a typical graph will have these probability values and timestamp of course it is not shown in this particular graph.

(Refer Slide Time: 33:12)




Now let us look at a few transition, few properties of this transition probability matrix. In fact, let us discuss this in the next class, thank you very much, we will continue in this discussion and about, study about the property of the transition probability matrix, thanks.