Software Engineering
Prof. Shashi Kelkar
Department of Computer Science and Engineering
Indian Institute of Technology, Bombay
Lecture - 32
Estimation - I

In the last two sessions we have considered project scope and project time. In both these areas we had a subprocess about estimation. In the first instance it was about estimating the effort involved in doing the job and in the second instance it was about the time required to complete the job. Thus, estimation is a very important activity in all the knowledge areas but in particular for the scope and the time management there are certain techniques which are very useful.

To begin with, just a few preliminaries like the estimation process is aimed at reliably predicting various parameters associated with making a product. It could be the effort, the cost, the time or the quality and the outcome of these particular processes is known as the estimates. The estimates really found the basis of agreement between all parties concerned.

For instance, there has to be an agreement between the customer and the supplier about the amount of the effort involved, the time required and the cost. Similarly, there has to be an agreement between the top management and the project manager about what resource is required, when, in how much quantity and of what type. Last but not the least, even within the project the project manager has to worry about giving the time, effort and other requirements for each individual program or doing a test or writing documentation.

Estimation basically forms the basis of agreement between all concerned. When you are doing estimation for software projects there are several problems you can unique to software projects. The first one of course starts with the software itself is a unique aspect or unique particular product which is very difficult to sort of it is an ethereal product. Then it is always one of rather than of mass production job because the mass production of software really is not considered as software development at all. This particular project also involves very large amount of human effort. So the intangibility of the product, one is of the product and large human involvement itself makes it unique.

Then we have problems with customer requirement. In most of the software development project the customer requirements are not clear and they gradually evolve as the project progresses. This poses its own problems in terms of estimation. And often we hear of features creep whereby what was the amount of time, money or effort that we started with the project the project ends up consuming more effort, takes longer time and obviously also delivers more coat and coat, the bulk of the product that was originally planned. We have yet another particular problem. We say lines of code is one way of estimating the size of the product then the definition of lines of code itself is not unique. Different

people need to look at different aspects before they can agree on what constitutes a line of code. We look at little later in detail.

Another particular misconception is that when you are doing software development estimation you are only taking of developing software but there are so many other aspects like training, documentation, convergence of data which are really very different from software development. So these particular types of activity need to be estimated separately.

Another unique thing about the software project is that technology. The technology changes very fast and we have many problems. First, use of new technology, you may not have people who have had prior experience in using that technology. Second particular part is that you may not have the appropriate metrics for doing the estimation with new technology and the platform itself may undergo changes while the development is still in process. So in that particular thing it is very difficult to estimate things. Last but not least you have the constraints like pressure of time, the development method etc.

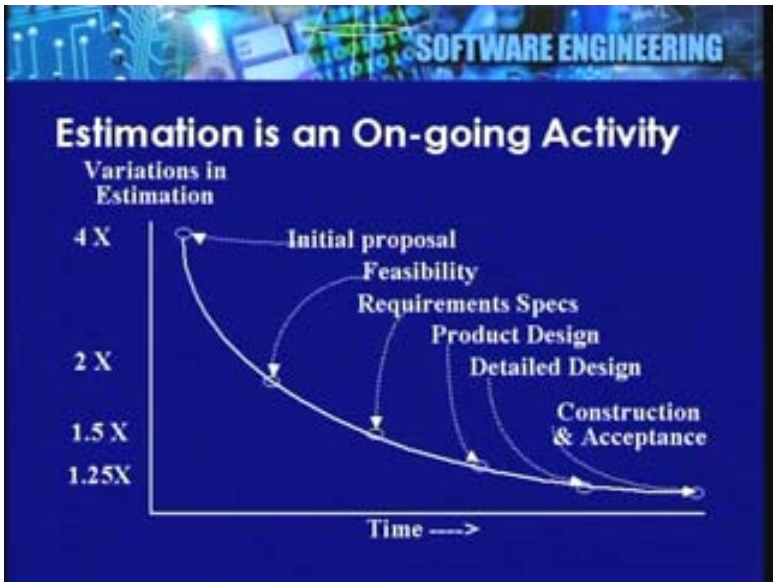
Sometimes the customer may want you to do the development in a particular way. The customer may want you to use particular standards or the customer may insist that the development be done on a pre-identified platform. So, in this particular way the estimation for software development activity is little different than making estimates for other engineering products. Then we have other particular aspect of productivity.Once you know how much of work is to be done you need to convert it into the effort and for that you need to have some kind of metrics about productivity. But here again we have problems, there are some aspects which are product related. We will see these in more details later on.

Then there are some productivity aspects which are machine and the platform and the environment related. There are yet others, when it is simply put, you may or may not get the most experience or training kind of people for doing the job and similarly the person may or may not have a prior exposure to technology or may not want to work on a project and so on and so forth.

And lastly project related; now this is the one which is largely in your control but you still have a choice of what particular methodology to use, what lifecycle to define, what matrices to use and how to implement what kind leadership to provide. So if you strictly look at it only project related attributes associated with the productivity only those under the control of the project managers and with the others the project managers has to just live with. So use of development tools can also make a dramatic difference to productivity and if you are using the development tools it means a lot of involvement initially to buy the tools then to train the people to use the tools and then keep the people up to date and then the development etc is a very long and elaborate […7:56] kind of a procedure. So obviously here is one more particular thing that is wherever possible you may like to buy some components or packages in software and not completely reuse as we talk about it all the time. Reuse may dramatically coat and coat, change your productivity.

In software development you must also realize that there is development productivity and there is system productivity. The development productivity is associated till the code is made working and given to the customer and accepted by the customer. Whereas if you are look at the productivity or the entire life of the product, specially if you are looking at the maintenance of the particular software then the concept of productivity will change very dramatically.

(Refer Slide Time: 8:52)



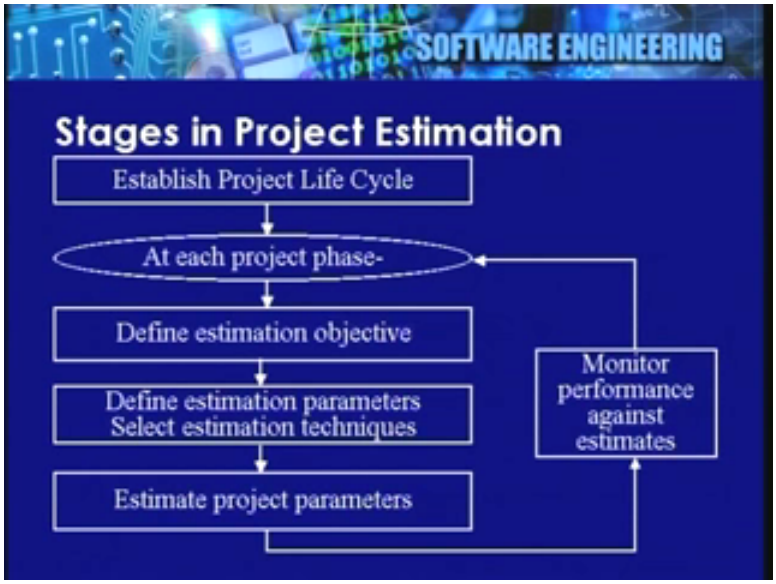
Now let us look at the next aspect which is of a particular kind. One of the things that you must remember is that the estimation is not a one time activity in a software development project; it is an on-going activity. It is done during each particular phase of the project. Estimates are done using different techniques for different objectives and they may have been done by more than one person depending on the seriousness or the importance of the accuracy of the estimate.

Take a simple analogy; if a father was to estimate the expenditure for his son's four years of engineering education he will make an estimate for the four years and after the first year he makes another estimate for the remaining three years and verify the estimates for the previous year and goes on and even after all the four years are complete he may try to look back and see whether his estimates were made well and may pass on this useful information either to his friends or colleagues or use it for his second son's education.

Thus, the estimation objectives if you look at it, as the time progresses in the project the information available for making estimates differs and more and more information becomes available as the project progresses, another information becomes available as the project progresses. This enables you to use different techniques for making estimation and we often want more than one approach for estimating so that we can cross check the total effort involved in doing the project. So this particular kind of approach is very important for you to realize that initially the variation in the estimations could be as much

as about 400% on either side and as you go closer towards the end of the project then the estimate is very accurate obviously is more like a post factor rather than the estimate kind of a situation. So we have this situation where we are using the particular technique. Therefore now we ask a question; how do you do the estimation in a software project.

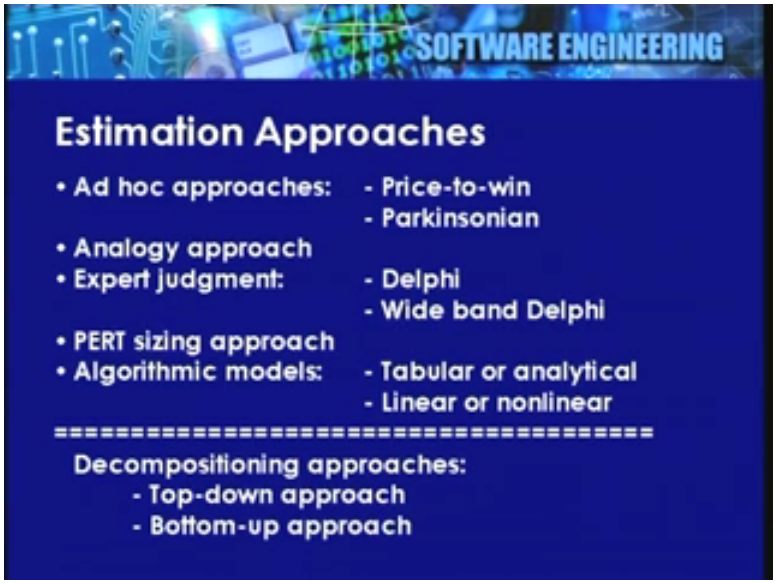
(Refer Slide Time: 11:25)



First and foremost the prerequisite for you is to establish a life cycle for the project. Once you establish the life cycle then you know the different phases in that particular project and for each particular phase you are going to set up estimation objective. Remember, the estimation objective for all the phases is not the same. For each particular phase the estimation objective is different. Once you have set the estimation objective the next thing is to establish the parameters which will help you in achieving the estimation objective. This includes setting up the procedures and all that kind of a thing. Therefore, once you have identified the parameters, the measuring devices, measuring approaches, the training people for doing the measurement, all those kind of things will have to be done.

So basically the project parameters have to be defined to an operation level then somebody will have to actually monitor and control whether our estimates are consistent with what we are expecting. So in this manner the estimation has to be done on an on-going basis. Now we come to the next particular question. What we do with this particular estimation? What are the techniques that are available for doing the estimation? First we have already seen that estimation needs to be done for various components of a particular project. The usual thing that we follow is “divide and rule” kind of an approach.

(Refer Slide Time: 13:06)



If you look at the bottom of the slide it says that we have to first have decompositioning approaches. The decompositioning approaches help you in taking a job and breaking it down into reasonably small components so that each component can be estimated separately if required and you can do it as a top-down approach or bottom-up approach and in real life obviously you take a combination of the two kinds of approaches the top-down and the bottom-up to come out with the components that need to be estimated.

Once we have these particular components then there are several techniques defined which will tell you how this particular estimate can be done. First we talk about Ad hoc approaches. The Ad hoc approach as the name says is really Ad hoc in the sense that there is no formal basis as to how this particular estimate was arrived at. But some of them work and especially if you have no other particular way of doing it then Ad hoc is better than not doing anything. Ultimately as it is said “the proof of the pudding is in the eating” you need to win the order. So these approaches can be used as a last resort when no other better approach is available. The biggest advantage that you have in Ad hoc approach is quick response time and the disadvantage is that the variation is very large and it is likely that at the end of the project there may be a lot of dissatisfaction being passed around.

Now the two distinct approaches used for Ad hoc estimation are Parkinsonian approach and the price to win approach. In Parkinsonian approach you find out how much resource is available at your disposal and deploy that entire resource for doing the development in the sense that if 200 people are sitting on a bench for this particular project then you say I am going to deploy all these particular 200 people and they are going to work on this particular project for a while. So it is more in terms of your ability to deploy the resource that you do this particular kind of a thing. But you always know the Parkinson’s Law that the work expends to fill the available budget and the time where at that particular token you might end up deploying far more resource than actually is required. So this usually

leads to over estimates. The price to win a project is exactly that you find out what at particular price you are likely to get the order, how you find that it is your business, […15:51……..] use any techniques that you write.

Basically find out how much the customer is willing to spend or when does he need the delivery of the product or how much the competitors are bidding or whatever. Make a bid so that you get the order. Here the assumptions are that the project is feasible and within the target and the contract is likely to be avoided if you bid for a particular cost, time and quality. The advantages apparently is easy and straightforward then does not need very expert assistance and you know that probably in advance that you are going to get the order. But this may lead to situations where you do under estimating of the resources required for doing the project and some amount of heart burning may come at the end because of bad quality product is likely to be delivered to the customer leading to poor reputation for a long time to come.

Another particular approach that you have next is the analogy approach. As the name suggests you find out that if you do not want inventory project and you want to do another inventory project you can compare the two and say like how much this one will cost, the last one cost so much. So here the projects that you are doing are similar, in simple words it is like Bata shoes stitching a size 6, 7, 8, or 9 shoes and the number of stitches per size is increasing in some kind of a proportion they can do some analytical calculations say that if the size 6 shoe takes so much then size 8 will take so much and so on and so forth. So what you are doing is you are relating the past projects to the proposed project. The comparison is made with one or more completed activities of similar nature and the estimates are made based on these comparisons.

While using the method you need to be sure that the information of the previous project is accurate because if the information regarding the previous project was bad then obviously your estimates will also go bad and the similarities are valid and amenable to extrapolation. Assumptions that were made for the previous project need to be known so that you can verify that your current assumptions are consistent with the earlier assumptions. This method is very useable as a back up or a verification method and there is no better way of doing it, this particular approach is also a very good approach for making estimates, the analogy approach.

Now, when you do not have any prior [18:36…] or prior knowledge then we resort to expert judgment in estimation. Here as the name suggests the expert is a person who is best in a position to give you an estimate and you really do not ask him how you got that particular estimate he just gives you a finger and you take it. So the estimates are based purely on the experience and the judgment of an expert. Now here if the projects are very large you may want to use more than one expert for doing the estimation and they somehow reconcile their estimates to come out with the final result. So this particular approach is emphasizing on the experts assessment of the situation and the past experience. In case the project that you are proposing is similar to whatever the expert has been doing, it will work well.

If you want to do a horse racing betting system any one who has done something resembling them would be a good expert for doing this particular kind of estimation. In particular you can also use it for handling exceptional situations or you can use it for re-enforcing the estimates with other methods or when little option is available you use this particular approach. The disadvantage is, the expert's capability to recall is likely to be in question. So estimates may end up being only subjective and may have personal bias like political influence and so on and so forth. So, in more than one situation you would like to use more than one expert for doing this particular kind of a job.

Now, when you have more than one expert and you want to reconcile their estimates then you have two broad approaches to do that. One is called consultative group consensus and the other one is non consultative group consensus. So there is a technique called Delphi which is used for non consultative group consensus techniques.

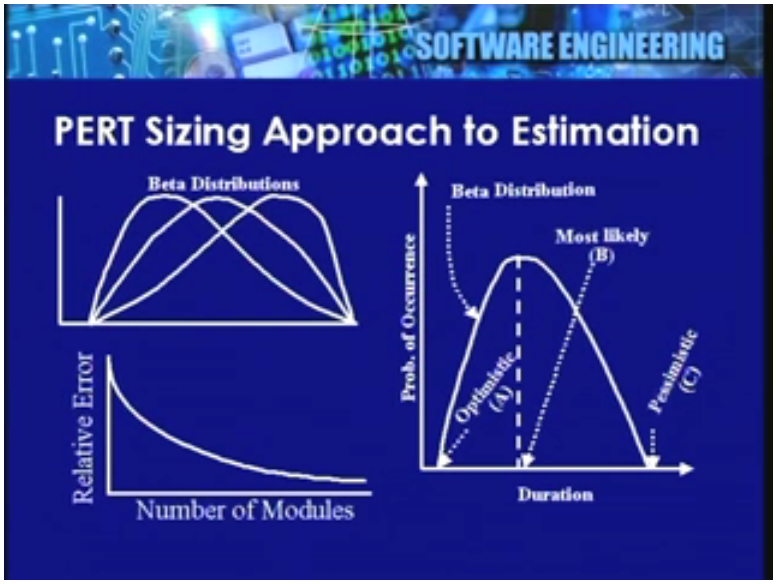
Here what you are assuming is you have a group of people who really do not need to come together or do not need to discuss because in case they start discussing there are likely to be other problems, there is no reason for them to sit together and work at the same time and come out and arrive at a consensus. Here every individual is free to give his own particular opinion and this opinion is shared with other experts without knowing whose opinion is that and the coordinator ultimately reconciles the estimation. So several experts are needed so it is a very expensive kind of a proposition and the advantage in this particular case being the experts may or may not be at one location and somehow you can make do with such situations.

There is a coordinator, what the coordinator does is initially explains the task to all the experts, give them the specifications for the particular task then each expert is suppose to do the estimates based on the information and past experience and send it to the coordinator, the coordinator then reconciles, summarizes all these particular things, in case you find that there is a wide variation amongst these particular estimates then the expert may be asked to give an explanation as to why in that particular experts opinion the estimate is very large or small and this data is circulated.

Looking at the estimates made by the other experts depending on their own past experience and the explanations given by the estimator, each estimator may like to revise one estimate and in course of time the consensus will be arrived at or the gap between the extreme estimates can be brought down to a reasonable or a pre-determined agreed level.

There is another particular technique called wide band Delphi technique which goes one step ahead of Delphi. It allows consultation among the experts, but a very minimal consultation. the major difference here is that you give the estimates and whenever there is a disagreement or only those particular parts of the estimates which are open to disagreement are discussed together by the group or otherwise those parts of the estimation which are agreed upon or fairly consistent with the various estimates that are given there is no need for these experts to come together. Many a times you have yet another particular issue that you need to look at.

(Refer Slide Time: 23:24)



We have another approach called PERT sizing approach. There is a very interesting distribution called Beta distribution. It is the uni-model distribution, it has its roots both on the positive side of the coordinate axis, it meets the x axis on the positive side. It is uni-model, it can be either left skewed or right skewed or symmetrical and what we can do is in case we are able to give estimates and make assumption that these estimates follow a Beta distribution then certain interesting calculations can be run. The human mind is incapable of conceiving the mean as a computed quantity in the mind. But you are often likely to know the mode as the most likely or the most frequently occurring kind of an estimate.

Now we have two problems; the human mind cannot comprehend mean but mean is amendable to further mathematical treatment whereas human mind is capable of perceiving mode but the mode is not amendable to mathematical treatment. Therefore, in case you have a situation where the human being can estimate the mode and you could compute from the mode the mean then it is like having the cake and eating it too. Now the PERT sizing approach is exactly aimed at this. If you make an assumption that human being can for a given particular situation give the minimum, maximum and the most likely values of the estimates then you can calculate the mean by using a formula like a plus 4, b plus c by 6.

So in case somebody says that how many lines of code will this particular specification get converted to and the expert may say it will not be less than 150 lines and it will not be in any case more than 400 lines and most likely be 200 lines and then you can say 150 plus 4 into 200 plus 400 by 6 is the mean time required for completing this particular project.

So if you look at the slide again you have this beta distribution as a unique particular property that if you were to give the most optimistic the most pessimistic and the most

likely estimate then there is some kind of mathematical relationship between the mean and the mode and of course between the variances and so on and so forth.

So if you use this particular approach then we ask a particular question, does anything in life follow a bitter distribution, the answer is no then we say how do we work on that and just to make a passing reference without going into the detail there is something called central limiting theorem.

Now we say, irrespective of the parent distribution the distribution of sampling in will always tend to be in normal and using this kind of a formula if you are able to take the project and break it down into components then the number of components is reasonably large then as the number of components increases the estimate that you get will tend to be more and more closer to the mean but also the mean of that particular estimation will follow normal distribution and you are able to do many of the other calculations with this thing.

The PERT sizing approach, the advantages, you need to go to a low level break up and that is why your estimates are likely to be accurate. The disadvantages that you need a lot of information for doing it that means you cannot do this particular kind of an approach unless you go fairly deep into the project. This technique is not usable in the early stages of the project. So once you have got these estimates for individual components then you can add up these particular estimates and make a total estimate for the entire project. Similarly, if you want you can find out a standard deviation and variance for the entire project, the formula for that being b minus a by 6 or whatever, the most pessimistic minus the optimistic divided by six is the reasonable estimate of the standard deviation and of course you know the principle that the variances are additive in nature and you can find out the total variations.

Incase you want to do probabilistic estimates then this technique will also tell you with how much confidence you can say that the project time or project cost or project quality whatever the parameter you are estimating with what confidence that particular parameter is achievable. This is a very useful technique. Now, beyond the PERT sizing kind of approach we have algorithmic modules. Usually people are enamored by this particular thing and they think that this is the end of the estimation which is not. But algorithm models do play a very important role in making the estimate.

Algorithmic models basically are based on the past data and often heuristic models are fitted to the data and without knowing why you know that incase you have a dependent variable you know how it is related to the independent variable. So the causal relationship is not known but the actual relationship you absorb is known. So you can use these algorithmic models for making reliable estimates based on the past data only a-priory you have a past data.

Now, incase if you do not have the past data you need to rely on the data collected by other people in industry and publish it. There are not too many publications, people do not often like to display or share their performance data but some industry models are

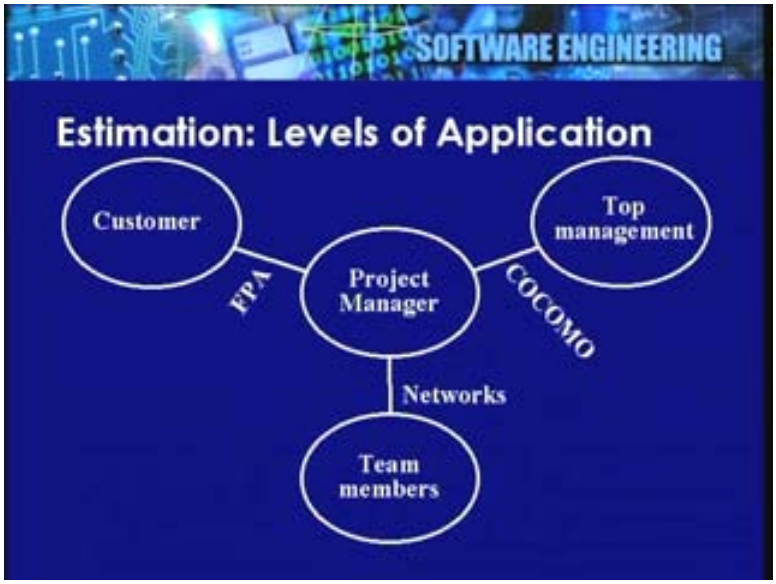
available and you can use those as a starting point and then modify those particular modules to fit your environment.

Now, in particular, some techniques like FPA are going to be very good for doing the estimation in phase of incomplete data at a very early stage in the projects life. So these kinds of estimates help you in providing envelopes for making some trade off decisions and so on and so forth. They are also useful during the subsequent stages for conforming or verifying the estimates that you may get by other alternative means. The algorithm models have disadvantages that some algorithmic models depend on estimates like line of code which can be given until you are fairly deep into the project. That means those particular models cannot be used in early stages of the project. Then in reality the controlled experiments are very expensive, difficult and rare. So we have to rely on the data “as is where is” take it or leave it. Then parameter values are also organizational dependent and you might find that from one extreme to another extreme you cannot get the same variations.

So, for instance, people working in software development of Tata Consultancy Services and people working in software development department of Life Insurance Corporation of India may or may not have the same parameter values for doing the development or maintaining the systems. Also, you are not in a position to handle exceptional situations using this kind of approach. This kind of approach is good when you have known situations. Suppose the technology has changed, it is very difficult for you to use the old data and use it for the proposed project. There are certain problems of this particular kind and those particular problems can be solved otherwise by using different approaches. Let us now look at making estimation for the software development part of the software project.

Many of the components like training, convergence all those for the timing less bypass them and look at how you make estimates for the software development project. Now you say whom do we make these estimates for?
It is a basis or agreement for all concerned. So let us take for instance three major stake holders; the customer, the top management of the project manager and the team member and what we need to do is to make estimates for these people so that we can work closely with them. Now the purpose and the details of estimation in all the three cases are not the same. Obviously from that point of view the estimates are done at different points of time and information available at that point of time is different. So obviously the techniques that we would use for doing the estimations will also vary.

(Refer Slide Time: 33:28)



First, look at the project manager and the customer relationship. Now, if you look little before the project started the customer needs a single estimate for the total amount of time and effort required for completing this particular project. That is basically your input to bidding. Now this aspect of project management of verifying the estimation is consistent with the marketing support that you need to provide for making the estimates.

Now, what are the characteristics of this, the customer has not paid you anything and the amount of data that is available with you is very little and there is a chance that you may not get the project. So the amount of time that you want to spend on this particular job also has to be reasonable but the accuracy should be fairly good. So the technique that we recommend for this particular purpose is called function point analysis.

Function point analysis is a technique which can be used at the very early stages of the project and it can work with very little data and give fairly accurate estimate, and it is also very easy to verify.

Now the next particular job is the top management. This estimation is required after you have got the job and now you are interested in this particular job being done. So if you were to go back to our analogy of the four years of the engineering education of a particular father then the total estimate for the education may be 4 lakhs over the four years but that does not mean that the father has to make available all 4 lakhs at once on day one and he needs to make a lakh of rupees available every year. But the same particular token, though the project estimate has been made the top management is not going to give all the resources to the project manager on day one.

For instance, when you begin your system analysis the top management will give you only those resources which are required for doing the analysis and at a later stage they will give you the resources required for design and then for coding and then for testing

and so on so forth. So you need a different type of estimation which is little more in detail and the technique that we will study is called COCOMO that is Constructive Cost Model by Barry Boehm. This function point analysis and COCOMO techniques are two well documented techniques in public domain which can be used without any hesitation by all concerned.

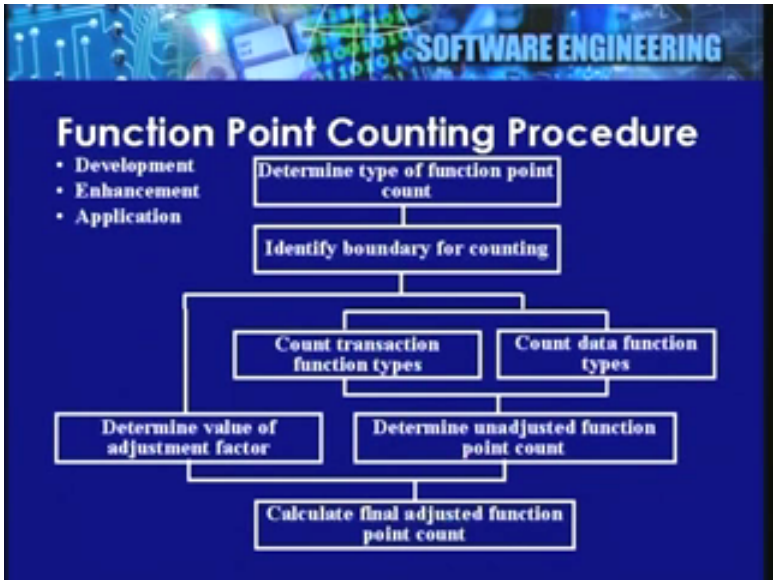
Now, once you got this particular resource from the top management you have another problem, you need to break it down into smaller components. For instance, the top management gives you 50 programmers to work for 6 months to develop an x number of programs, this is fine but what you really need to do is to estimate for each program the time and the effort for completing it. Ultimately we need to come down and say on what date should the program start, how much effort is required to do the job and when will the program be complete. So the techniques that we are going to use for doing the particular kind of a thing is application of well known network techniques like PERT and CPM.

Using these particular three techniques you are able to solve many of your estimation problems during the software project. Of course there may be lots of other techniques available which are instead of [n…37:25.] to the user but these are the techniques that are very well documented and available in the public domain. Now let us look at these particular techniques in detail. First of all let us begin with function point analysis.

What is function point?
Function point is the unit of the size of the software application. Remember, size of the application. Lines of code are another measure of the size of the application. So function point is not interchangeable, the two different measures is like if we are buying a shirt you can possibly go and buy it by the color size or you can buy it by the chest size both of them make an estimate of the type of thing that will fit in you. Similarly, lines of code and function point are two different measures of the size of the application.

Function point analysis is a relative measure of the functions as delivered by the developer to the users. It takes into account only those functions which are seen by a user. Typically to give you an example, incase you had to do normalization and because of that you have to do certain other particular processes and modifications to your procedures for updating the data on the database those kinds of things are not too visible to the user. So we say functions only as seen by the user, then we say which user, for all users the end user, the developers, the operators, the auditors the quality assurance and everybody, anyone who needs a specific function for performing his or her job then it is considered as a part of the development effort. So we say that the function points are counted around a single application boundary. So let us look at how this function point counting actually takes place.

(Refer Slide Time: 39:46)



First we need to determine what type of function point count we are doing. There are three types. The first is called the development function count. What is the development function point count? You are interested in developing application software from scratch. Then you have the enhancement function point count. Here what you are doing is you are making modifications to existing software which may involve adding modifying and deleting code. And the third is you may be interested in actually estimating the amount of or what is the size of a product that already exists.

Take a simple example, if two companies want to merge with each other and they would like to valuate their asset and they say yes the bank which is trying to merge with another bank says that we have so many programs and then you ask a question there that how much is this asset worth and then will you need to go and find out in some particular way. Therefore, one way of doing it is that, the software I have has so many function points, the three applications are different. The first is you are making an estimate for what is yet to be developed. Second; you are making modifications to what already exists and in the third case you are assessing or evaluating something that already exists. So first of all you need to look at the type of function point count.

Once you have got the type of function point count the next thing that you need to do is to identify the boundary. We have already highlighted that identification of boundary is central to function point counting. There is only one application boundary and the functions as you have seen by the users resulted in transactions going in and out of the boundary so we identify the boundary equation. Once we have identified the boundary the next thing that we do is we would like to start counting the function. There are two types of functions; one is transaction function and the other is data functions. So from our particular point of view you look at it first we count the transaction type function.

What is a transaction type function?

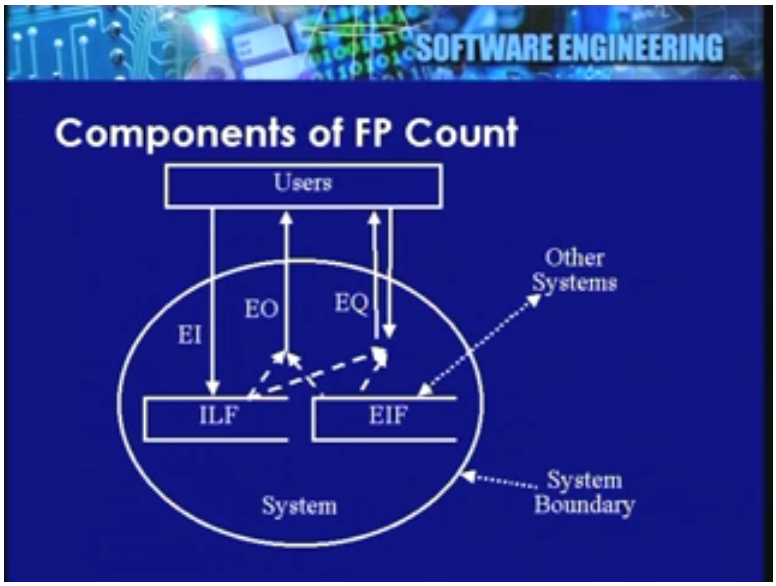
The transaction type functions are basically the inputs, the outputs and the enquiries. And we count if each one of them is simple, average, complex, and transparent and so on and so forth. Once you have done that then we would like to count the data function type.

What are the data function types?
Once you got the data in the transaction, once you have collected the data with the transactions or using the data for transactions this data must be kept somewhere so this data has to be stored within the boundaries of the application. So we count in what form this data is stored inside the application boundary and that is what we basically say that they are data function types. These values that we get, both these particular situations are basically what we call unadjusted function point counts after doing an appropriate modification. Each function can be classified again as simple, average, complex and depending on that you have a raw weight associated with each function and when you total that up you get an unadjusted function point count.

Now you also know that from one system to another there could be differences and some tuning up may have to be done accordingly. So the function point counting approach provides for some adjustment factors. There are fourteen adjustment factors and each of these particular factors can be rated from absolutely irrelevant to most essential. These particular adjustment factors are additive in nature. You add them up and you get a total adjustment factor which is in the function point jargon known as degrees of influence and the adjustment factors are known as general system characteristics.

Once you got the adjusted factor then you need to multiply the unadjusted function point by the adjustment factor to get the final function point count or what we in normal sense call only the function point count. So you end up with the function point count for that particular source. This is the procedure for computing the function points.

(Refer Slide Time: 44:26)



When you start counting the function point we have users and you have a system and the system has a boundary and the users do three types of transactions; the inputs, the outputs and the enquiries. People often have difficulty in what you call distinguishing between the outputs and the enquiries. We can only explain to them on a little lighter side. Some of the things that come to your mind is that the output is some kind of a batch mode and enquiries is something online. Both these particular notions are farthest from the truth.

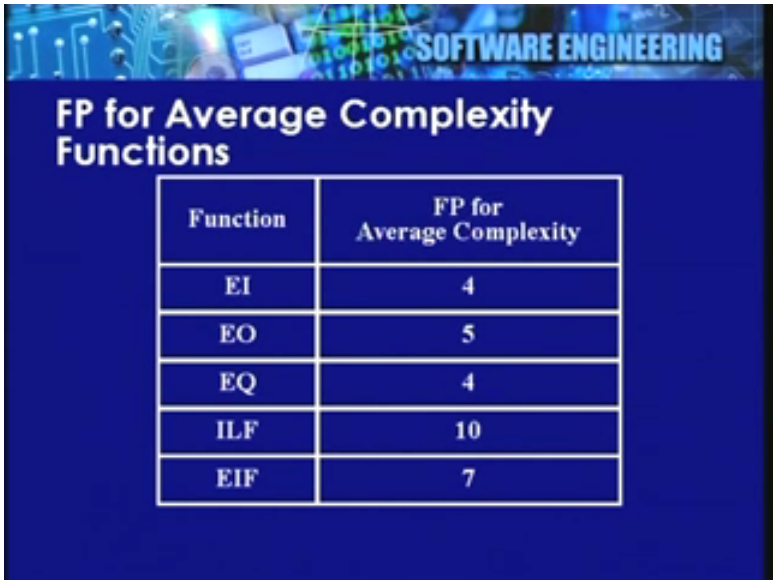
You can have an online output and you can have a batch output similarly you can have online enquiry and you can have a batch enquiry. So how do we distinguish between the two and there is one particular way of separating it. If you are physically retrieving the data that is stored inside the system then usually we call it as the enquiry. It is like looking up in just saying an answer like there is a simple example; when you say, can you give me the list of all the people who have not paid their bills for the last three months, it would be an enquiry, it does not involve any computational effort. But if I was to ask your another particular question; can you tell me how much is the total outstanding attributed to people who have not paid for three years, three months or something like that then this involves doing a computation and then this would be in a simplistic sense classified as the output. There are more elaborate explanations as to what the output is and what the enquiry is. These are the transaction function types. Then we have a simple question.

Once we get the data in the system what we do with this particular data, we need to keep it somewhere and the place where you keep this particular data is known as internal logical file. So the data coming from the input can go to one or more of the ILFs and similarly one ILF may receive data from more than one input. Hence there is a many to many relationship between the inputs and the ILF.

Now there is one more interesting thing that you must keep in mind. Sometimes we are interested in using data that is generated by the other people but which is of use to us. Take a simple example; incase you are developing a hospital management system and you are interested in classifying the diseases and you would like to use international code of diseases published by the World Health Organization. Then this particular file of data is not maintained by you but used by you. So EIF is basically files which are developed and maintained by other applications but used by our application.

Take another example, if personal department of a company maintains an employee file and if the payroll department of the company uses the employee file for generating the payroll then the employee file is a ILF for the HR department and employee file is a EIF for payroll department. So the EIFs usually come from other particular systems. Another example we can give is; if you want to go to an international airport then the international airport authority system interacts with other airlines systems, and in case you are looking at the time table or the flight arrival or departure board then the ILF's of different airlines systems are sent as a EIF to the international airport authorities system. So, in this particular manner we can do the calculations and say that we calculate the unadjusted function point counts and the adjusted function point counts.

(Refer Slide Time: 48:48)



Now the average weightage is given here. Tables are available for simple functions and complex functions but for average functions you can say that average input is equal to four unadjusted function points, average output is 5, enquiry is 4, ILF is 10 and EIF is 7. Now, if you look at this particular table just for your information if you had a simple function then the data would be a simple input is equal to 3 unadjusted function points, a simple output is 4 unadjusted function points, simple enquiry is 3 unadjusted function points and simple ILF would be 7 and simple EIF would be 5. By the same particular token if you have a complex function then a complex EI would be 6, a complex EO would be 7, a complex EQ would be 6, a complex ILF would be 15 and complex EIF would be 10.

Now, by looking at this particular data all of you must realize that when we come to enquiry every enquiry is a combination of an input followed by an output. So when you are looking at an enquiry how do you classify an enquiry as simple, average or complex is basically you find out whether the input part of the enquiry is simple, average or complex and the output part of the enquiry is simple, average or complex and take the larger of the two as the weightage to be given to the enquiry.