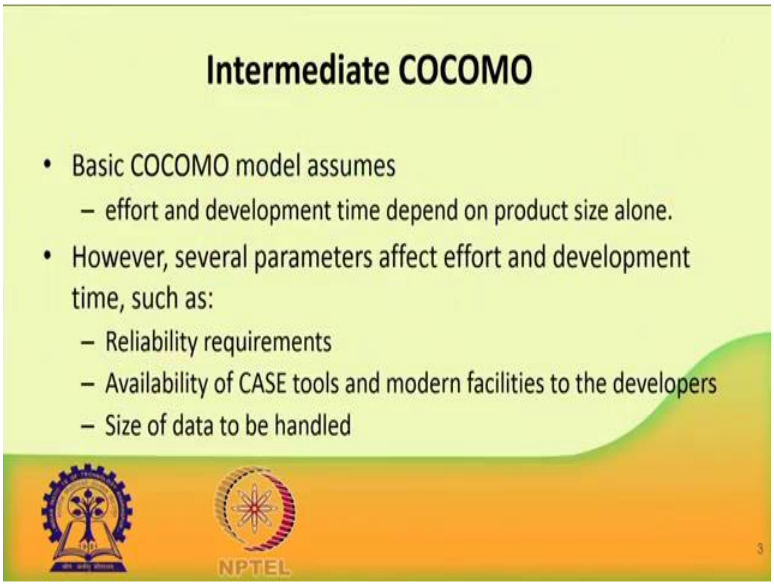
Software Project Management
Prof. Durga Prasad Mohapatra
Department of Computer Science and Engineer
National Institute of Technology, Rourkela

Lecture -22
Project Estimation Techniques (Contd.)

(Refer Slide Time: 00:27)

Good morning. Now, let us take up the other cost estimation technique. Last class we have discussed about, basic COCOMO model. Now, let us take about other two types of COCOMO that is a intermediate COCOMO and a complete COCOMO.

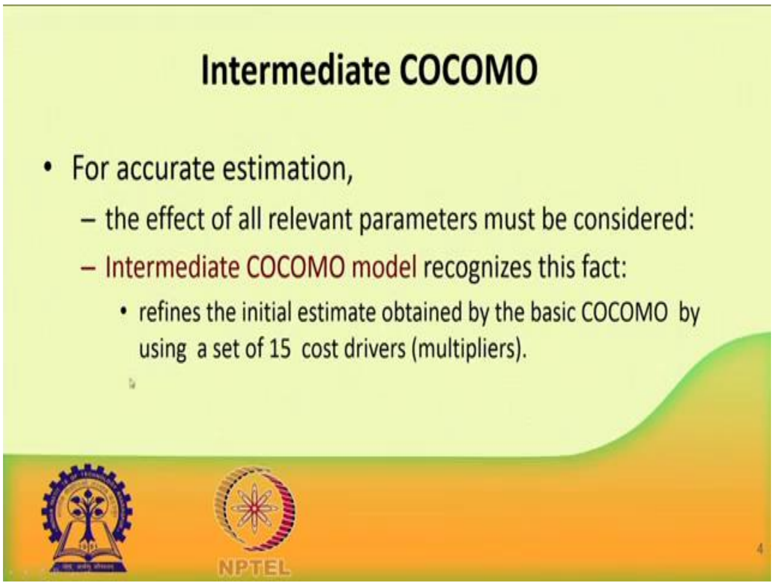
(Refer Slide Time: 00:31)



See ah there are little bit disadvantages of basic COCOMO, the basic COCOMO model assumes that the effort and development time, they depend on the product size only. However, you will see in the modern project developments several other parameters they also effect effort and development time, such as reliability requirements, you might require that your product should be highly reliable.

You may you are using several modern available case tools or some modern programming facilitates to the developers definitely the usual reduce the efforts, and this size of data to be handled it is quite large. So, these requirements or these parameters may affect the effort and development time. So, these have to be addressed in the COCOMO model. So, in order to consider these issues, the basic COCOMO model has been enhanced.

(Refer Slide Time: 01:36)



So, the newer COCOMO model we call as the intermediate model. So, for accurate estimation and to consider the effect of all such modern relevant parameters. So, a new COCOMO model little bit advanced COCOMO model was proposed that is known as a intermediate COCOMO. The intermediate COCOMO model recognizes the fact that, all these relevant parameters they have been addressed.

So, first you have to make the initial estimate, then the intermediate COCOMO refines the initial estimate which is obtained by the basic COCOMO by using a set of 15 cost drivers. So, first you have to prepare the initial estimate using COCOMO, then this initial estimate they can be it can be refined by using a set of 15 cost drivers or multipliers. Now, let us see what could be the possible what 15 cost drivers those are applicable for intermediate COCOMO.

(Refer Slide Time: 02:39)

So, before going to the 15 list lets taken example that how these cost drivers or these what multipliers, they are affecting the effort or the cost. See for example, if somebody is using modern programming practical practices rather on using the traditional programming practices, then definitely the effort required will be less. So, hence the initial estimates obtained by the basic COCOMO they have to be scale downwards.

Similarly, suppose you for you are developing a real time system a real time application for which there is stringent reliable requirement, it must not fail there should not be any delay, they are should not be what the deadline must be met. So, there are several other reliability requirements on the product. Then what will happen, you have to put much more effort, what does it mean that the initial estimate what you have prepared, what you have ah obtained that has to be scaled upwards. So, in this way you can consider the other parameters how they are affecting, the estimation of the or the initial estimates of the effort and cost.

(Refer Slide Time: 03:50)

So, then what you have to do as I have already told you intermediate COCOMO uses 15, what cost multipliers or these cost drivers, you have to rate the different parameters on a scale of one to three. So, these different parameters they can be rated on the scale of one to three depending on these ratings what you have to do, you have to multiply the cost driver values.

So, how may see all the 15 cost drivers or the these parameters may not be applicable for every system might be 1, 2 or 10 whatever that you know which parameters, which cost drivers will be required for your project. Find out or rate them using a scale of one to three. Then multiply the values of these cost driver these cost drivers with the estimate which is obtained by using the basic COCOMO. So, this product this multiplication the result of this multiplication will give you on more a more accurate estimate. So, this is the objective of intermediate COCOMO.

But please remember that even if I am saying here that the rating should be made in between one to three, but in some cases the value of the parameters or the cost drivers may be also less than 1. For example, when you are using modern programming practices instead of using the traditional programming practices, then what will happen; what will happen this effort required will be much less, much less means you have to go downward. So, we have to in that case the value of the cost driver will be less than 1;

might be 0.9 something that will I will give you in the table. So, what will the exact values for this cost drivers.

(Refer Slide Time: 05:27)

So, let us see how these intermediate COCOMO proceeds, it takes the basic COCOMO as the starting point using the basic COCOMO prepare the initial estimate. Then identify what are the attributes they are affecting the cost and development time for your project; such as personnel attributes, product attributes, computer attributes and the project attributes. So, all those 15 attributes are categorized into these four categories. So, these attributes are expected to affect the cost and development time of your product.

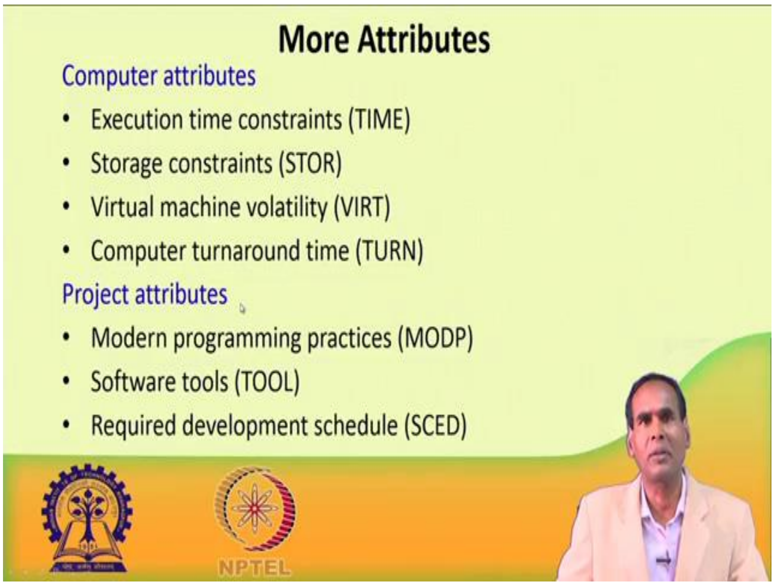
Then what we have to do? We have to multiply the basic cost that you have obtained by the basic COCOMO. Then you what multiply the basic cost by these attribute multipliers or the cost multipliers which either may increase or decrease, this initial basic cost.

(Refer Slide Time: 06:25)

Now, let us see what are the basic the attributes I have told there are 15 attributes categorized into four categories such as, personnel attributes, product attributes and other two are, I have already told you; there are four attributes will see. The four attributes are product attribute, project attribute, here personnel attribute and what computer attributes. So, these let us see all those 15 attributes under the these different arrays.

Under personnel attributes you will see that the analyst capability, virtual machine experience, programmer capability, programming language experience and application experience, these attributes are coming under personnel attributes related to the what related to the personnel working for the project. So, these are the codes there may be used for these attributes, like for analyst capability you may use the code ACAP, etcetera. So, for product attributes these are the product attributes, like reliability requirement, database size, product complexity these are the attributes related to the product that we have developing.

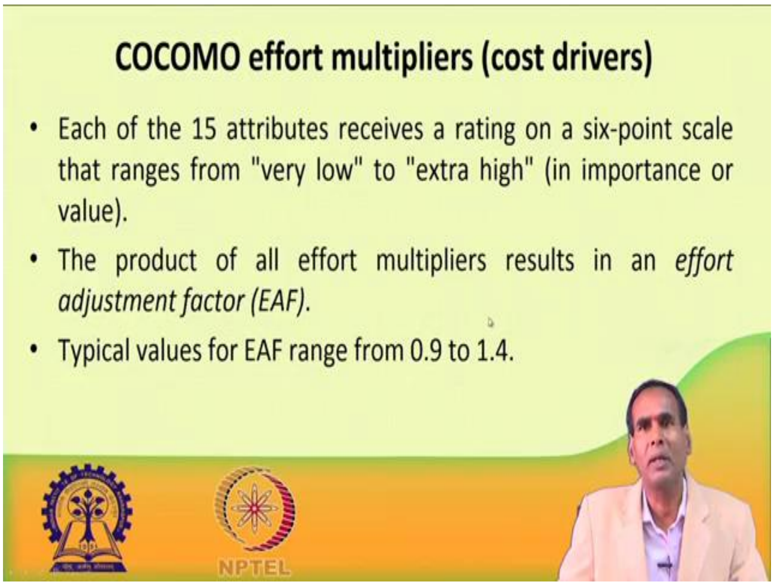
(Refer Slide Time: 07:37)



And then computer attributes you must consider, what are the execution time constraints, what are the storage constraints, the virtual memory volatility, the computer turnaround time, so these factors these attributes also may affect the what effort and cost of your product. And finally, project attributes or the environmentally attributes you must consider, project attributes also known as environment attributes.

So, what modern programming practices you are using? Like whether are you using software reengineering, reuse, reverse engineering, etcetera or object oriented programming, so all the modern programming practices, then that will make your effort and cost much less. Then what kind of software tools, automated software tools are using if I doing something manually, but if some tools are used then obviously, the effort required will be less. Similarly, required development schedule, so what is the schedule that is required; whether it is very tight schedule or this is flexible schedule, this will be also affecting the effort and cost. So, these are the 15 attributes that you must consider to get on up what more accurate estimate of the effort and cost.

(Refer Slide Time: 08:48)



So, COCOMO effort multipliers as I have already told you, these are also known as the cost drivers. So, each of the 15 attributes they receive a rating, they receives each of the 15 attributes receives a rating on a six-point scale it will, we will use a six-point scale that will range from very low to extra high, I will show in importance or in value. So, this you may consider as the what is the importance, if it is importance is very low or low or medium or high or extra high like this or may use. So, for these different categories like difference just from very low to extra high, you may use the values one to three or any points, I will give you the table for these values.

So, then after for identifying which attributes are affecting your product, then see it can be categorized as whether very low or high etcetera, then find out the corresponding value from the table, I will show the table in the next slide. Then you have to find out the multiplication the product of all such effort mutiliers, and this multiplication will give you a what term which you call as EAF or known as the Effort Adjustment Factor.

Then and this effort adjustment factor may range from 0.9 to 1.4. Then what we have to do, we have already found out the basic the initial estimate by using basic COCOMO, multiply this EAF or the effort adjustment factor with the base initial estimate obtained by basic COCOMO which will give you the refined effort or the ah this will be the more accurate ah estimate for the effort.

(Refer Slide Time: 10:31)

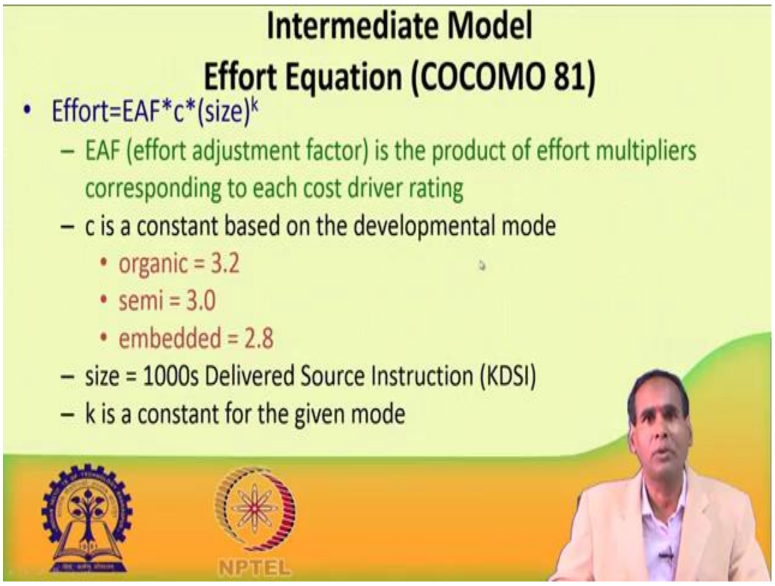
Cost Driver	Very low	Low	Nominal	High	Very High
Required reliability	0.75	0.88	1.0	1.15	1.40
Database size		0.94	1.0	1.08	1.16
Product complexity	0.70	0.85	1.0	1.15	1.30
Execution time constraint			1.0	1.11	1.30
Memory constraint			1.0	1.06	1.21
Virtual machine volatility		0.87	1.0	1.15	1.30
Computer turnaround time		0.87	1.0	1.07	1.15
Analyst capability	1.46	1.19	1.0	0.86	0.71
Application experience	1.29	1.13	1.0	0.91	0.82
Programmer capability	1.42	1.17	1.0	0.86	0.70
Virtual machine experience	1.21	1.10	1.0	0.90	
Programming language experience	1.14	1.07	1.0	0.95	
Modern programming practices	1.24	1.10	1.0	0.91	0.82
Use of software tools	1.24	1.10	1.0	0.91	0.83
Development schedule	1.23	1.08	1.0	1.04	1.10

See these are the 15 cost drivers as I have already told you, there is a what six-point rating very low, low, nominal, high and very high; so it is I think 1, 2, 3, 4, 5, the five so here it is written six, so then this should be actually five, please correct it. On a rating of the five-point scale that will better and here it is five there are five what ratings are shown. So, like required reliability is equal to 0.7 when it is very low, when the required reliability is very high it should be 1.40.

And you can sees in some cases, the value will be less than 1. For example, if the product complexity is very low you may assign 0.70, these example I was saying that if modern programming practices are using and if what it is very high, then you see if it is high it is less than 1 nine 0.91; if it is very high it is 0.82. You can see the nominal values of all the cost by drivers the average values, the nominal values are all ones and based on that if it is high, it will be more than 1.

And normally here what is happening for some of the what ah some of the parameters like analyst capability, etcetera; if it is very low, it will be what greater than greater than 1. So, in this way you can use these values of the cost drivers to get the refined estimate.

(Refer Slide Time: 12:06)



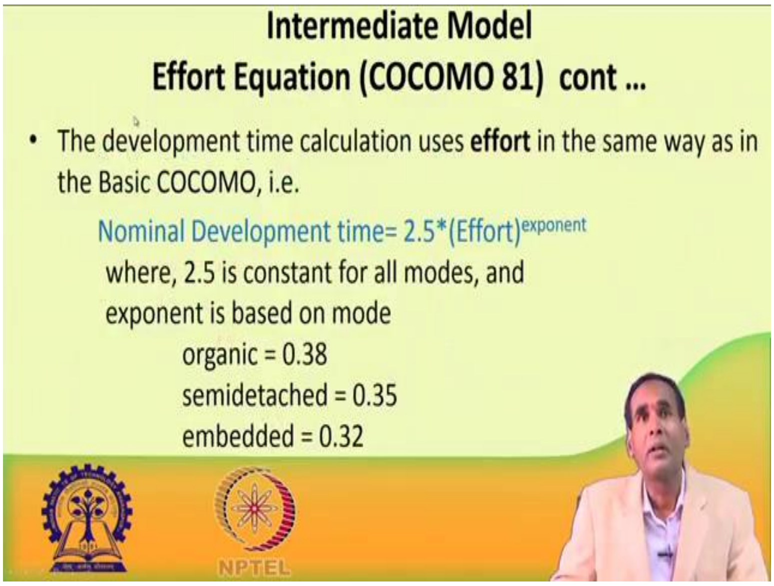
So, let us see what will be the equation for this intermediate effort, it intermediate model for calculating or estimating effort. So, the effort equation can be stated I like this that

$$\text{Effort}=\text{EAF}*c*(\text{size})^k$$

so c almost similar to the basic COCOMO only here EAF has been added. So, the EAF is the what this adjustment factor. So, you can see EAF is this Effort Adjustment Factor which is the product of what, the effort multipliers that you have considered for your product corresponding to each cost driver rating.

And c is a constant based on the developmental mode, we have already seen. There are three development modes, may be organic, semi detached and embedded and same values we have already used in what earlier; so for organic it is 3.2, for semi detached it is 3.0 and for embedded it is a 2.8. So, the size is equal to what 1000s delivery source instructions or KDSI and k is a constant for the given mode. So, in this way you can estimate the effort using this intermediate; using this intermediate COCOMO model, you can estimate the effort using this equation.

(Refer Slide Time: 13:27)



So, similarly you can also compute the time the development of the time calculation, uses the effort which you are getting ah by using this equation using the EAF. So, this effort you can use in this earlier equation that we have already discuss for the basic COCOMO. The development time calculation it uses effort in the same way as in the basic COCOMO that means, nominal development time will be

$$\text{Nominal Development time} = 2.5*(\text{Effort})^{\text{exponent}}$$

Here this effort is the effort that you have obtained by multiplying the what effort adjustment factor. So, this is the you can say refined effort or this is the revised effort that value you can put here, this will give you the development time.

Where, 2.5 is a constant for all the modes may be what organic semi detached or embedded. And this exponent is based on the mode that you are using, for organic modes this is 0.38, for organic products this is 0.38, for the semi detached products this is 0.35, for embedded products the value of exponent is 0.32. So, by using these two formulas; one for effort which is the multiplied with EAF and these one for this what nominal development time by using this formulas, you can estimate the value of the effort and cost ah by using this intermediate COCOMO model.

(Refer Slide Time: 14:55)

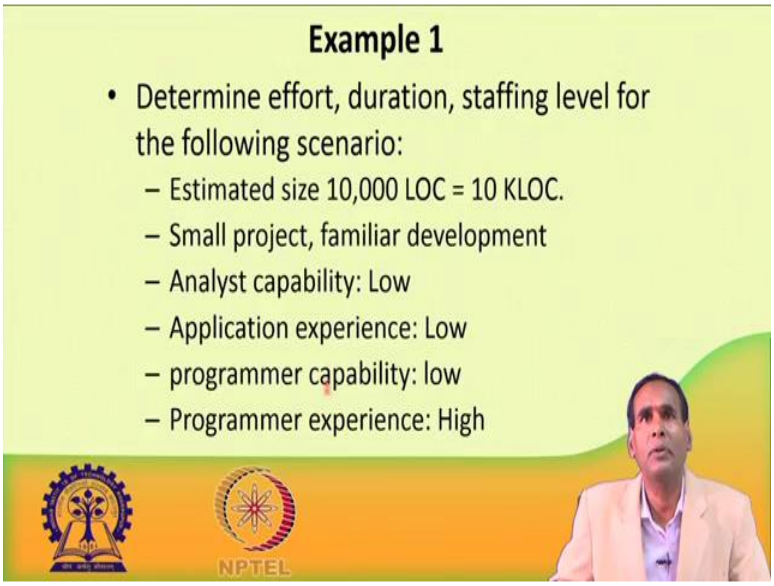
Now, let us take a small example we will consider how to use this cost drivers or the effort multipliers. So, the example said that you are considering the what analyst capability, the capability of the; the capability of the analyst and then you want to find out the estimate for the effort.

You can see, it is given that see depending upon the capability of the analyst, you can rate this capability as very low or low, nominal, high or very high. So, the if it is very low, you can see it is 1.46 and low is 1.19 you can see in this table. See what is analyst capability, what is analysts capability; yes it is analyst capability, you can see analyst capability. Yes here, analyst capability if it is low, it is very low 1.46, if it is low 1.19, if it is nominal 1.0 and if it is high it will reduce 0.86 and it is very high again it will reduce 0.71. So, like this you can find out these values.

So, find out the values of the multipliers in this what we have already shown in the table. Now, suppose the initial estimate of the effort is 32.6 and suppose the analyst is having very high capability, analyst capability is high, analyst suppose the analyst capability is high that means, the value 0.80 multi this is the what only one cost driver is there, where multiplying this. So, this will the value of the effort adjustment factor that is 0.80. So, the value of this effort adjustment factor will be multiplied with 32.6. So, this will give you roughly 26.8 staff months.

So, in this way so we are getting the adjusted nominal estimate or the we are adjusting the nominal estimate for the effort. So, the what refined effort or the adjusted effort is coming to be 26.8 staff months by using the effort adjustment factors 0.80. So, in this way you can use the values of the cost multipliers, to find out the refined value of the refined estimate of the effort.

(Refer Slide Time: 17:21)



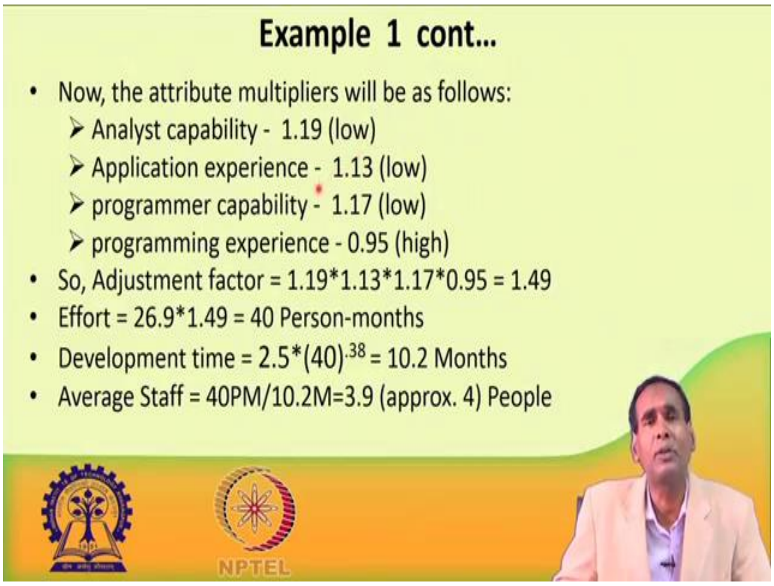
Now, let us take another example. So, determine effort, duration, staffing level for the following scenario. So, suppose according to what suppose you have estimated the size of a product is to be a 10,000 LOC which is equal to ten KLOC. And it is if it is small project, familiar development is there. And you are estimating that; you are expecting that or it is given that the analyst capability is low, application experience is also low. Programmer capability is a high and programmer experience is what and programmer experience is also high.

(Refer Slide Time: 17:59)

So, what you do, you take we want to find out the cost effort, etcetera. It is needed to produce see from the problem, you can see that needed to produce a program of size 10 KLOC. Since is a small project and familiar development, it can be treated as a organic model. So, effort we will use what, we will first use these effort using what using the basic COCOMO.

So, if will take the basic COCOMO it is equal to what standard formula that this value of c is 2.4, size is 10 to the power the exponent value will be 1.05 for organic products, this will be 20 roughly 26.9 person-months. Development-time against a forward formula 2.5 into effort to the power what the exponent for organic product this is 0.38, so roughly it will give 8.7 months. And hence average staff will be how much, effort divided by development time so this is equal to roughly 3 people you will get. Now, what you will do, we have to get the refine value using the cost multipliers or the cost drivers.

(Refer Slide Time: 19:05)



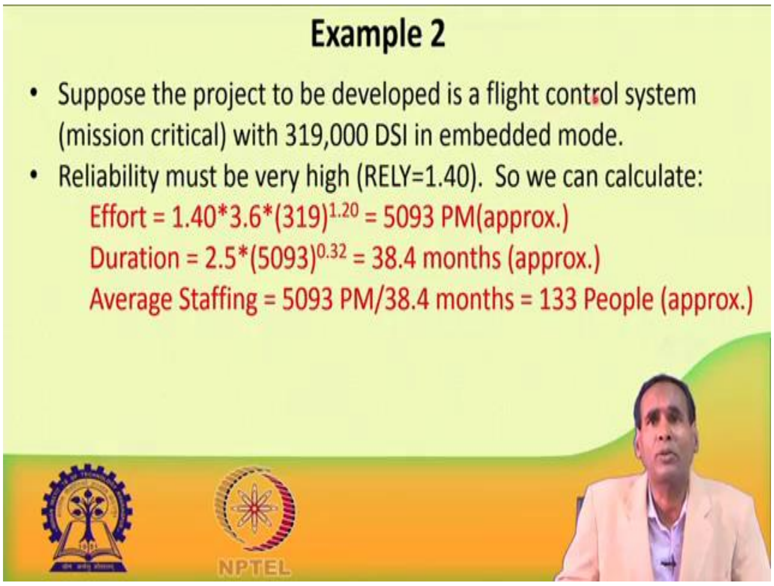
Now, the attribute multipliers will be ah as follows, these values we can get from the table that I had provided. So, analyst capability is low; low you know it is 1.15. You have application experience it is also low that is 1.13 and programmer capability it is showing programmer capability, perhaps will have to see here. Programmer capability is also low, here something just one minute. Yes, programmer capability is also low here.

So, programmer capability is low and this value you can take programmer what will low is 1.17 and, but programming experience is high. So, value will be the programming experience is high; it will be less than 1, so 0.95 already given in the table. So, the adjustment factor will be how much multiplication of all those values. So, this you will give you EAF Effort Adjustment Factor this is coming to 1.49.

So, the revive effort will be completed as the basic estimate of 26.9 into the EAF – Effort Adjustment Factor 1.49, this will give you 40 person months. Now, the development time will be what? 2.5 into 40 not these 26.9, I have to take the revised value that is 40 to the power exponent for organic product this is 0.38, so it will be roughly 10.2 months.

Now, the average staff; so these are all the what refined values by taking into account the values of the cost multipliers. So, average staff will be now required 40 by this is equal to 4 people 3.9 or approximately 4 people. So, in this way we can use these values of the cost drivers or these multipliers to refine the basic estimates more accurately. We will quickly take another example.

(Refer Slide Time: 21:02)



Suppose, we have to develop a project for a flight control system, which is a mission critical with 319000 DSI in embedded mode. Of course, this a flight control system this is a very risky project. So, it is what is a very complex project treated as embedded, it is a type of embedded product.

The reliability is required to be very it should be very high and for reliability very high values 1.40. So, now we can calculate the effort is equal to how much, the basic effort first we have to compute or the initial estimate by using the basic COCOMO; so 1.40 into the value for embedded value embedded case, the concern value c is 3.6 and the DSI is given 319 kilo what LOC and the exponent value for embedded mode is 1.20, this is equal to 509 percent month approximately.

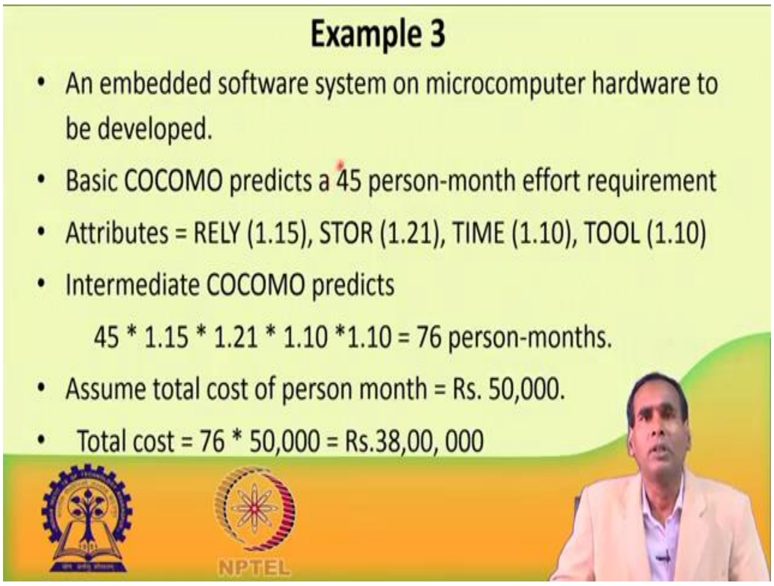
Duration will be again strait forward formula, this will be 38.4 months and average staffing will be equal to how much effort by duration, so coming to be roughly 133 people approximately. Now, we have to use the cost drivers, you can see that only one cost driver is given, only one attribute is given, so I have to directly multiply with this.

So, and so I have to directly multiply of course, we have already direct so already I am sorry, that we have already in the while calculating the effort already I have taken into account this what reliability, because in according to basic COCOMO we should get only this much 3.6 into 319 to the power 1.20.

And if we will apply intermediate COCOMO, I have to take the value of the effort adjustment factor; since only one cost driver is given that is reliability which is equal to what 1.40 for very high. So, I have multiplied here 1.40 into 3.6 into this, so we will get 5093 percent month, this is the refined estimate for effort.

Similarly, duration you can use this what refined effort the what revised effort that is 5093 to the power 0.32, it will give you the revised duration. And this average staffing will be equal to this revised effort or the refined efforts by this duration which is equal to 133 people for approximately. In this way, you can use the cost drivers to refine the value of the initial estimates.

(Refer Slide Time: 23:30)

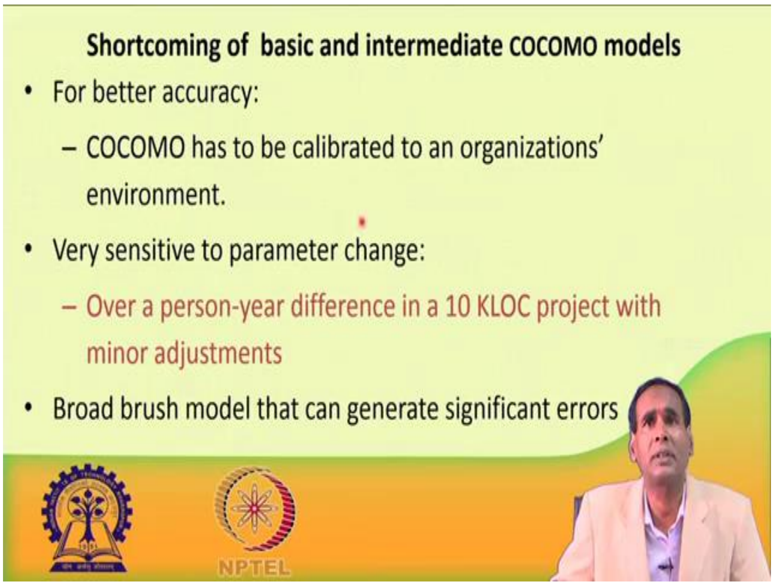


So, similarly quickly we will take the other example and embedded software system on micro computer hardware is it has to be developed. Basic COCOMO it predicts that 45 person-months effort is required. So, here there are four attributes are required, so four cost drivers you have to find out; one is reliabilities, storage, time and tool. The values are already given and suppose these values are all already given for this embedded software system.

So, intermediate COCOMO can now predict or estimate the effort as such what will be this, so already basic COCOMO gives 45 person months, so 45 into effort adjustment factor. How we can find out the effort adjustment factor? By multiplying all these what the values of the cost drivers, so for cost drivers there values are modified are multiplied. So, this will give you 76 person months.

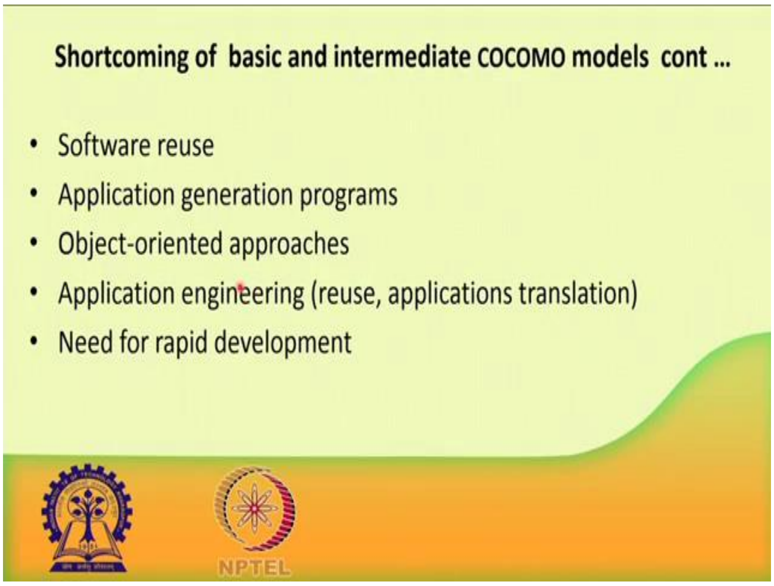
So, assume that the cost per months person month that we have to pay each rupees 50000. So, the total cost for developing this embedded software system products will be 76 into 50000 which will be equal to rupees 3800000. So, in this way you can use the embedded software, you can use the intermediate COCOMO to estimate or to refine the basic estimates.

(Refer Slide Time: 24:45)



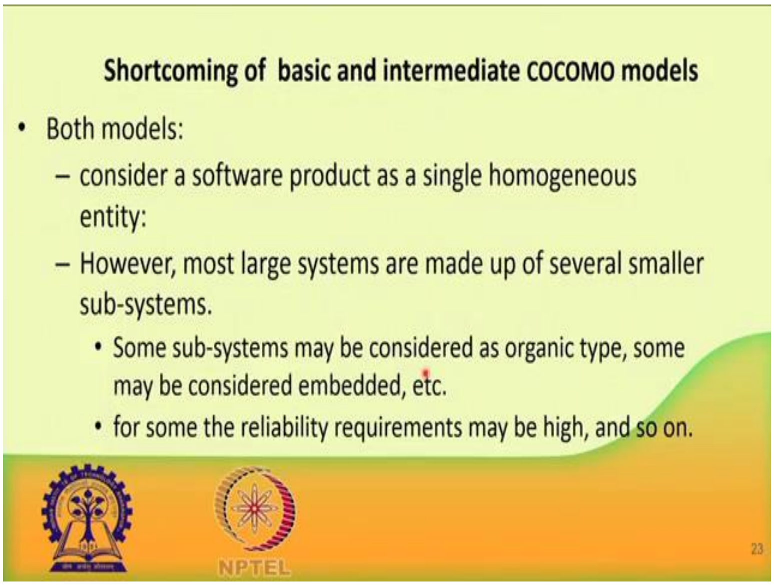
So, now let us quickly see about the shortcomings of basic and intermediate COCOMO model. For better accuracy COCOMO has to calibrated to an organizations environment, where is it is very sensitive to parameter change if you just make some my, if you just do so minor adjustments, minor changes over a person-year difference may arise in case of 10 KLOC project. It is a broad brush model that can generate many significant errors, these are some of the drawbacks.

(Refer Slide Time: 25:13)



Similarly, as you know this in this modern what today's software reuse is there, application generation programs are there, object oriented approached are used and application engineering such as reuse, reverse engineering, application translation, etcetera, they are and so there are need for rapid development. So, this basic and intermediate COCOMO they do not considered these modern practices. So, one so these are the some of the drawbacks the basic and intermediate COCOMO models do not consider these modern programming what practices. So, there is need to go for another model.

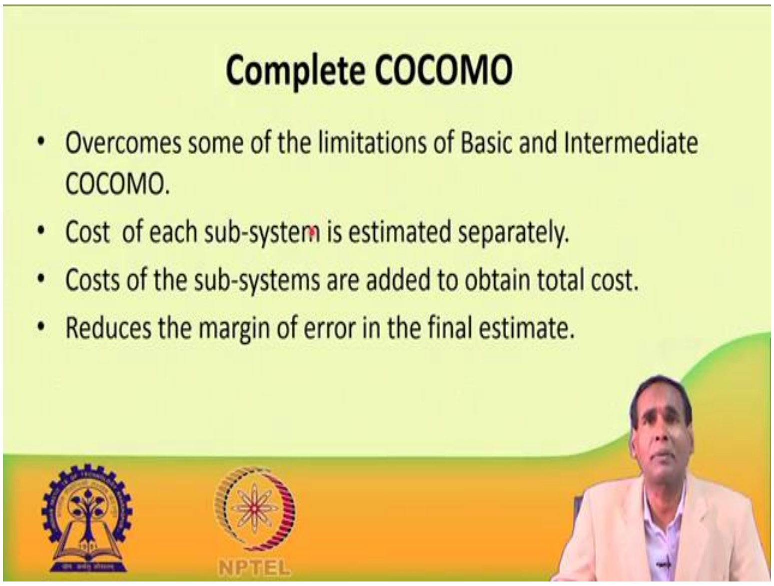
(Refer Slide Time: 25:47)



So, we will see still then another important drawback is there, which say that this both the models. Basic model and intermediate COCOMO, they consider a software product as a single homogenous entity as if it is considered only one type of what components. But however, most large systems they are made up of several smaller sub-systems, where some sub-systems may be considered organic time type, some sub-systems maybe of what embedded type, some sub-systems may be of the semi detached type ok.

And in some cases reliability, some of the systems may require high reliability requirements or at the some of them may require very low reliability requirements things like that. So, your basic and intermediate COCOMO cannot handle all this things, because it considers all the sub components are treated as say similar, because the what software product is considered as a single homogenous entity.

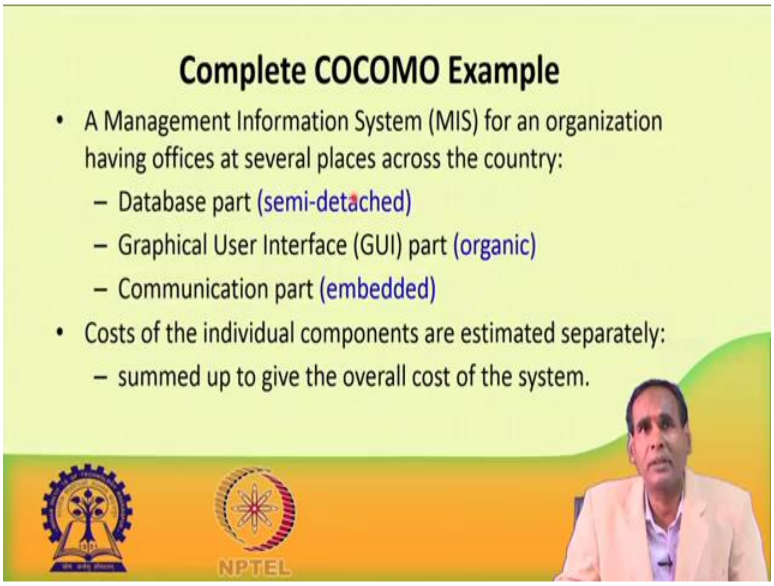
(Refer Slide Time: 26:47)



So that is why a new so another advanced model of COCOMO it has come into existence that is known as complete COCOMO or detailed COCOMO. It overcomes some of the limitations of a basic and intermediate COCOMO. How does it work? Here the cost of each sub-system is first estimated separately using this what basic COCOMO or the intermediate COCOMO. Then cost of the sub-systems they are added to obtain the total cost. So, because the as I have already told you, some of the sub-systems might be organic, some of the sub-systems might be semi detach, some of the sub-systems might be what embedded, in some substance they reliability might be very high.

So, you have to estimate the cost of each sub-system separately, individually, then the cost of the substance they are added to obtain the total cost. This will reduce the margin of the error in the final estimate; obviously, we will get a less error or this will reduce the margin of there are in the final estimate.

(Refer Slide Time: 27:44)

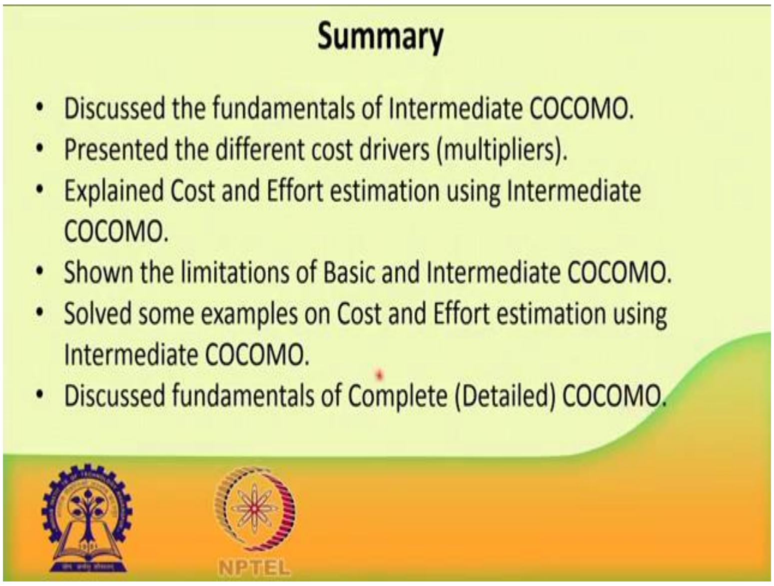


Now, let us quickly take a small example. A management information system suppose we have to develop for an organization, which is having offices at several places across the country. So, then this will be a very complex project, this is not an homogenous what project, this consist of different types of sub-systems.

Like one part may be may contain the data base part, you may treat as a semi detached system. One part one component might contain development of graphical user interfaces; obviously, this will be treated as organic component or organic product. Similarly, since the MIS has offices at different locations in the country. So, the these different offices they have to communicate. So, you have to develop also a module for communication. So, this module will be treated as embedded.

So, now what we have to do, compute the cost or estimate the cost for these three components, which are of three different types by using either basic COCOMO or intermediate COCOMO. Then what you do, add up the cost of the individual components ok, first estimate the cost of the individual components separately for each of these what components. Then you add up these the cost of these individual components, which will give you the overall cost the total cost of the system. In this way, the complete COCOMO or the detail COCOMO it works.

(Refer Slide Time: 29:13)



So, finally we have come to the summary we have discussed the fundamentals of intermediate COCOMO, we have presented the different cost drivers or multiplies; 15 cost drivers and their values we have seen. Then also we have explained the cost and effort estimation using intermediate COCOMO, we have shown some of the limitations of basic and intermediate COCOMO. We have solved some examples on cost and effort estimation using intermediate COCOMO, we have also discuss the fundamental concepts of complete COCOMO or detail COCOMO, this is the summary that we have seen.

(Refer Slide Time: 29:47)

And we have taken from these books; these texts.

Thank you very much.