Principles of Engineering System Design
Dr. T. Asokan
Department of Engineering Design
Indian Institute of Technology, Madras

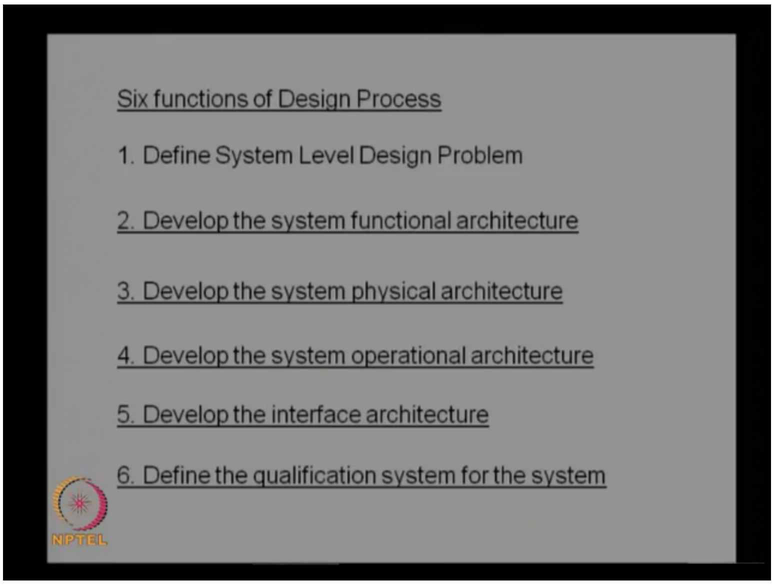
Lecture – 12
Physical Architecture Development

Dear friends, a very good day and welcome back to another session on systems engineering. In the last few classes we discussed about the requirement analysis of engineering systems originating requirement developments and functional decomposition. We found that using the requirement analysis, and the originating requirements documents, it is possible for us to identify the functions required in a system, or from the top-level function we will be able to decompose these functions into small functions or the smallest functions, and then prepare a functional hierarchy for the system.

The next logical step is basically to go for the design of physical system. So, this is basically developing a physical architecture for the system, where we try to convert these functional blocks into physical elements. We will try to see how we can identify corresponding physical elements for the function, or how do we do a mapping of the function to physical elements which will satisfy the functional requirement of the system.

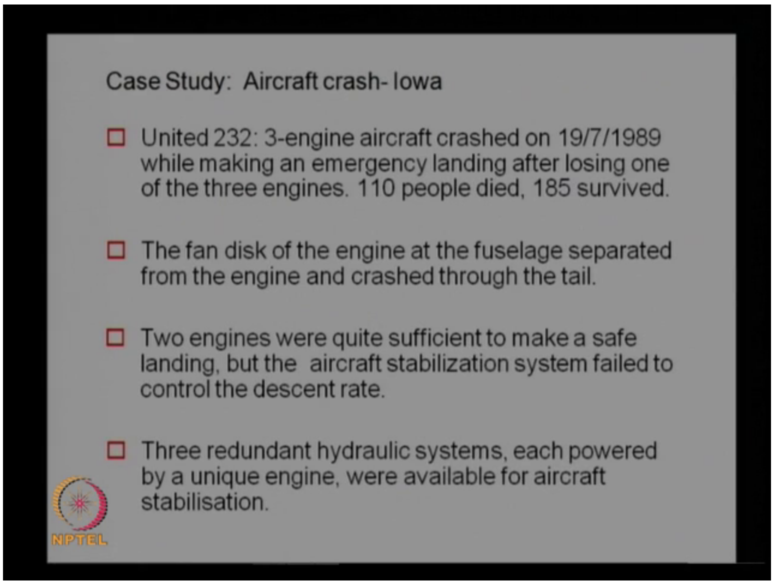
So, in this chapter of physical architecture development, we will try to look at the functional decomposition and then the conversion of this function decomposition into a physical architecture of the system.

(Refer Slide Time: 01:31)



As we shown in the previous lectures, the out of a 6 functions of the design process we are completed the system level design problem and the system functional architecture developments, and the next task is basically the system physical architecture development.

(Refer Slide Time: 01:50)



To explain the importance of physical architecture development, I will just explain case study basically of an real incident where the failures in the physical architecture development led to the failure.

So, using this case study I will emphasize the importance physical architecture. It is not only the a customer requirement functions, but are, but other functions like, fault tolerance and identifying the physical component for fault tolerance plays a vital role in the system development. And this case study is a perfect example to show that how the physical architecture development over look the importance of having a fault tolerance system, and how this led to the failure of a important system especially in aircraft.

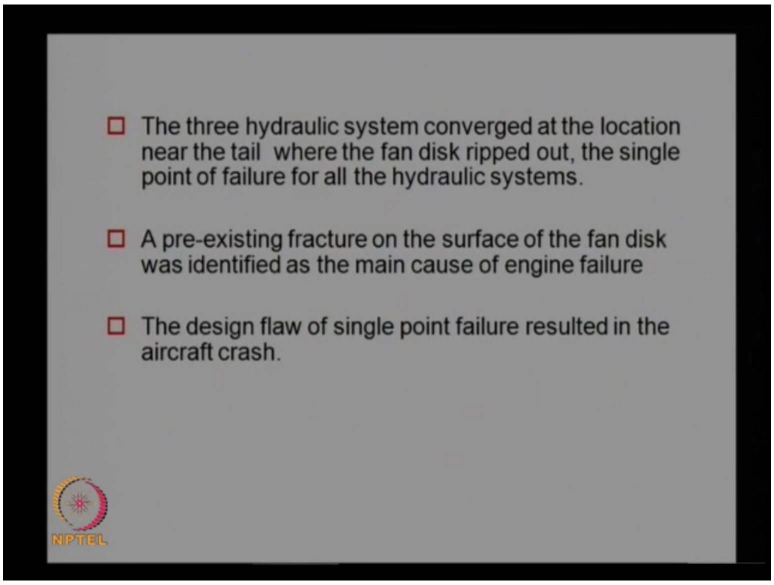
So, this actually is a case study from an aircraft in Iowa. The united the aircraft 232; which was a 3-engine aircraft a crashed on 1989 while making an emergency landing after losing one of the 3 engines. And in this case one 10 people died and 185 survived. Most of you know most of the aircrafts are designed with lot of fault tolerant system, and in spite of all these aircraft failed.

As I mentioned it was a 3-engine aircraft, and even if one engine fails or 2 engine fails it is possible to bring the aircraft to do a safe landing, and the pilots are being trained to overcome this failures in the system. And there are sufficient a fault tolerance in the system to overcome such emergencies. What actually happen here was the fan disk of the engine at the fuselage separated from the engine and crashed through the tail. So, that was the immediate cause for the failure. The fan disk of the engine separated and it crashed through the tail of the aircraft.

But that one engine failure was not problem because 2 engines were quite sufficient to make a safe landing, but the aircraft stabilization system failed to control the descent rate. So, what actually led to the crash was the stabilization system failed. But how the engine failure caused stabilization failure was an important aspect of the system descent. Or and we look at that system descent, we will see that it is not only necessary to look at the individual system we need to look at the overall system, and then see what are the possibilities of failure and analyze it.

So, here the even with the 2 engine the system stabilization system failed, how did happen was; there were 3 redundant hydraulic systems, each powered by a unique engine. Which were available for aircraft stabilization? So, there were 3 hydraulic system. So, I can see this was a redundancy in the system, one hydraulic system is sufficient to stabilization, and there were 3 hydraulic systems, each one powered by separately by each engines. So, even if one engines fails and one hydraulic system fails there were lot of redundancy is available, and it was possible to still control the aircraft. But the these 3 redundant hydraulic systems could not stabilize the system.

(Refer Slide Time: 04:59)



Because the 3 hydraulic systems converge at the location near the tail where the fan disk ripped out, the single point of failure for all the hydraulic system.

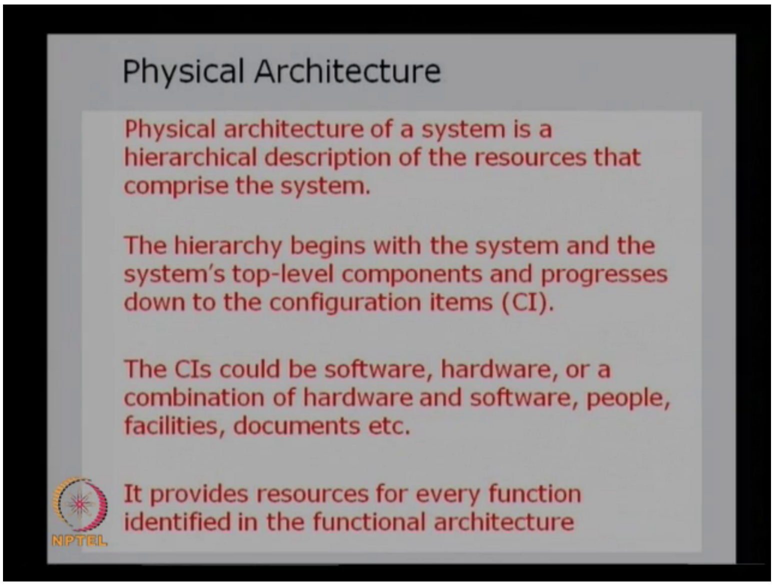
So, the point where all the hydraulic system outputs where converging at one location and the fan disk exactly ripped at that point and block the hydraulic system. That was the cause because even though all the 3 hydraulic systems are working perfectly, and there were 2 engines powering the hydraulic system. The hydraulic power supply could not reach the control plane actuation points, because this was a single point where all the 3-hydraulic system converge to the give the supply.

So, this was a single point failure in the system. And this is one of the important point to look at when we provide the fault tolerant system we need to avoid the single point failures and then make enough redundancy in the physical architecture, to make sure that such single point failures do not occur in the system.

So, the physical architecture development is very important or it actually plays a vital role in identifying such situations and then eliminating the possible causes of failure. So, in this case there are pre-existing fracture on the surface of the fan disk was identified as the main cause of engine failure. So, that was the main cause because the fan disk had some problem. But the design flaw of single point failure resulted in the aircraft crash. The main reason why the aircraft crash was the single point where the all the hydraulic systems where converging, and from that point onwards it was a single point failure situation.

So, this was the reason why actually it resulted. So, in the architecture development we will look at ah the fault tolerant systems, and then how do we provide the necessary redundancies in the system to avoid this kind of failures of engineering systems. So, let us look at the physical architecture development from the logical steps forward from the functional architecture. So, as we progress we will look at the redundancies and then how do we provide the physical system to have a sufficient redundancies in the system, and avoid such single point failures.

(Refer Slide Time: 07:14)



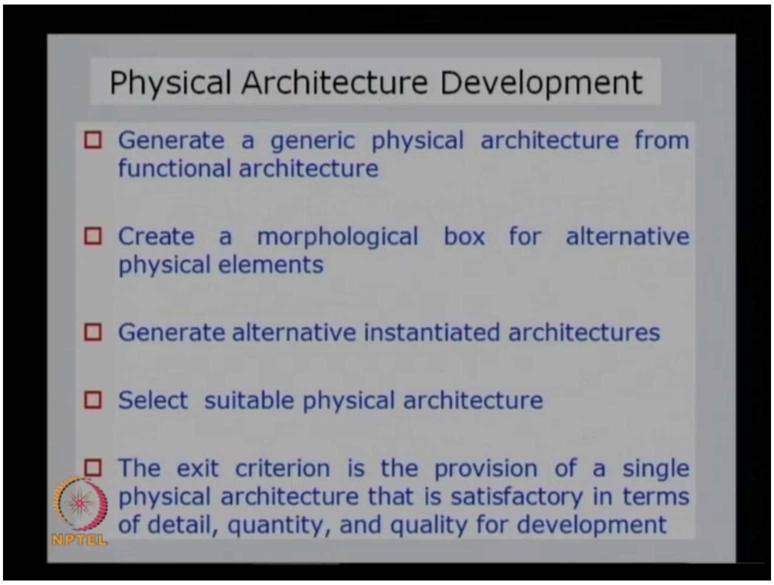
So, here let us discuss about the physical architecture, and what is physical architecture? So, physical architecture of a system is a hierarchical description of the resources that comprise the system. So, basically, we a look at the physical resources, which actually provide the functions in the system. So, architecture is basically we a write down the hierarchical structure describing each physical resource or the physical element which provides the necessary functions in the system.

The hierarchy begins with the system, and the systems top level components and progresses towards down to the configuration items. So, we start with the top-level component or the top-level assemblies or sub assemblies, and then we move towards the sub assemblies and components and finally, to the configuration items. Configuration items are basically the software hardware or a combination of software and hardware people facilities or documents. So, any of these could be a configuration item. So, we will list down all these items in a hierarchical fashion, and then complete the physical architecture.

So, in the physical architecture by looking at the physical architecture, will be able to tell what are the components being used there, what are the assemblies, what are the sub-assemblies, and what are the other items which actually comes as configuration items in terms of people facilities as well as the documentation. And then we will be able to identify which are these components provide the functions or which function is mapped to the particular component, or which are the component which provide multiple functions, or which are the multiple functions satisfied by a single component; all these things you will be able to identify from the physical architecture.

So, it provides the resources for every function identified in the functional architecture. So, that is the basic idea we you have the functional architecture, and from the functional architecture we will try to identify the resources for every function.

(Refer Slide Time: 09:11)



Now, how do we develop these physical architecture? It actually starts with the functional architecture. As I mentioned it is a logical step next to the after developing the functional architecture.

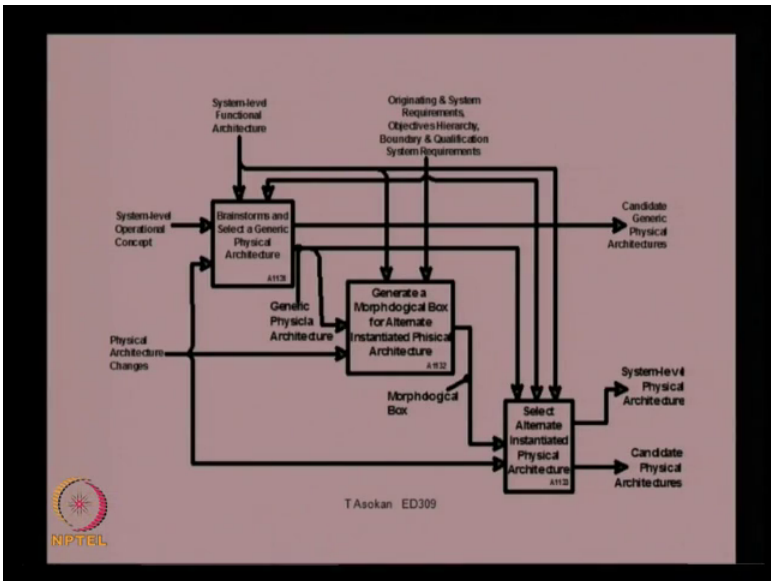
So, we develop a very generic physical architecture from the functional architecture. So, generic physical architecture is basically a conversion of the functions into component, but we do not identified the exact component. Here we just write the generic name or the generic way of representing that function in terms of a component. So, that is the generic physical architecture, without giving any details of the components used and then from that generic and physical architecture in order to get the physical elements. We create a morphological box for alternative physical elements. We will discuss about these in detail how do we do the morphological box or how do we develop the alternatives.

But morphological box is a tool for developing the alternatives for physical component. So, we identify one generic component in the generic physical architecture, and then we develop or we identify all the generic component, and then develop morphological box for these components; where we will have many alternatives for that component and then choose the component. So, that is the use of morphological box where we have alternative physical elements. And then using these alternative physical elements we will generate the alternative instantiated architecture.

So, instantiated architecture is basically an architecture a physical architecture where we identify the component, and then write down the component instead of the generic physical name of the component. So, here it is more specific and that is why it is known as instantiated architecture. And from these instantiated architecture we will be having a many choices for he here, and then based on the our requirement will select a suitable physical architecture. So, we go through the these steps, we will start with a generic physical architecture, and then we go for a morphological box to identified the alternatives, and using the morphological box we develop instantiated physical architecture, and using the instantiated architecture will identify a decide physical architecture for the system.

So, that is the process. And the exit criterion or when we decide or how do we choose a physical architecture is basically the provision of a single physical architecture; that is, satisfactory in terms of a detail quantity and quality for development. So, we look at the many possible options in the instantiated physical architecture. And then see whether it actually satisfy the in detail in quantity and quality of the development requirement because this has to be developed and we need to make the system. So, we look at the a details of the instantiated physical architecture, and then see whether the quality and quantity are actually matching with the requirement. And once it is satisfied I will go for that particular architecture and choice the physical architecture for the system.

(Refer Slide Time: 12:11)



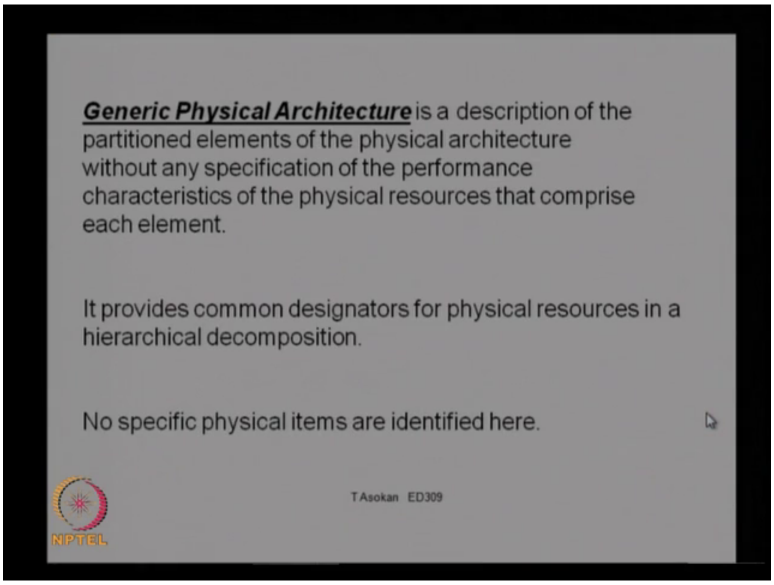
So, that is how we developed the physical architecture of a system. So, this is explained using the IDEF 0 diagram for different process. So, as you can see here. So, this is the first level well we do the brainstorming and select a generic physical architecture. So, in this generic physical architecture of course, we need to do little bit of brainstorming here from the functional architecture to convert the functional architecture to a generic physical architecture is bit tricky, and requires lot of a discussion and understanding of the functions and then finding out the generic elements for the generic architecture.

So, we need to do bit of brainstorming. So, the team members will sit together and then look at the functional architecture. And then see how do we convert that in to a physical architecture. So, for these the input will be the system level functional architecture, and then the system level operational concept will be used in identifying the generic physical elements. So, these are the 2 inputs, and then will be using this inputs and that will be used in the brainstorming session, and will be getting a generic physical architecture here. And this generic physical architecture along with the input from the system level functional architecture will be used for generating the morphological box for alternate instantiated physical architectures. As I mentioned the morphological box will give you the alternative elements for the generic elements identified in the generic architecture, and using this morphological box we can actually have alternate instantiated arc physical architecture.

And from there we can actually use these inputs from the instantiated physical architecture and of course, will be using the input requirements as I mean the operational concept as well as the functional architecture, and then the requirement will be using selecting the alternate instantiated physical architecture which actually satisfy the other requirement of the system, and that actually gives you the system level physical architecture. And there would be many candidate physical architectures based on the instantiated physical architecture. And the choice as I mentioned it depends on the quantity quality and the practical the feasibility of implementation, and developing the system.

So, this is the general process of developing the physical architecture. As you can see there would be many interactions and input and output between these, and whenever there are some changes will be go back and then get changed in the physical architecture generic architecture as well as the instantiated physical architecture will go through few iterations to make sure that we are actually reach in a stage where we can choose one architecture for development, that is the way how the physical architecture development is happening.

(Refer Slide Time: 14:54)

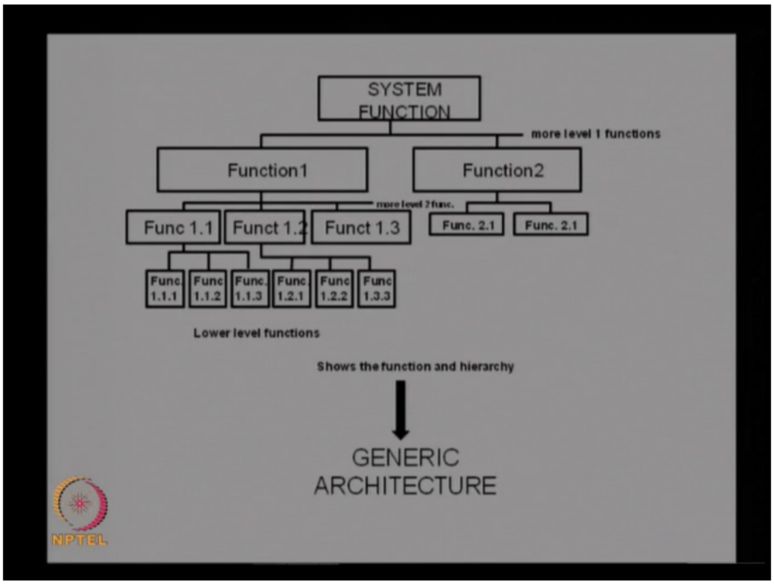


So, let us go into the details of how we actually develop this. As I mentioned the generic physical architecture is a description of the partitioned elements of the physical architecture. Without any specification of the performance characteristics of the physical resources that comprise each element. So, here we actually partition the elements as from the functional architecture we identify the elements, and then partition them and write down them as in a hierarchical way a similar to the functional hierarchy.

So, using the functional hierarchy will develop a physical hierarchy of the elements. But in this case, we will not give any performance characteristics of the physical resources. So, it will be more like very generic an names. So, no performance specifications will be mentioned in the architecture. So, that is the generic physical architecture though it actually provides common designators for physical resources in a hierarchical decomposition. So, it is or more like very common or general designator for the elements or no specific components will be identified. So, the no specific physical systems are identified here in the generic physical architecture. So, this is basically as a starting step for the physical architecture.

So, converting the functional architecture to a generic physical architecture by giving the generic name for this components without specifying their performance, or the physical element used in the architecture. So, that is the generic physical architecture.

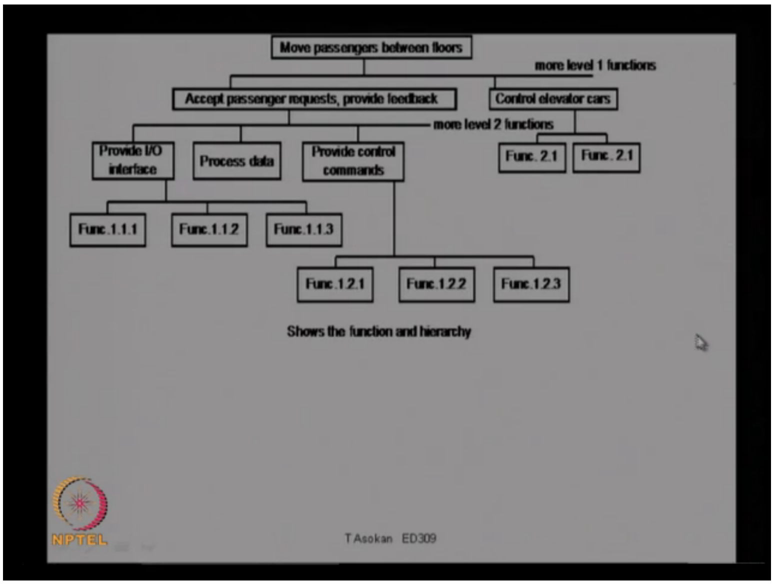
(Refer Slide Time: 16:22)



So, as we can see this we have seen the previous lecture. So, this is the functional architecture. The functional architecture we write down the top-level function, and then the sub functions and then we decompose them into many levels, depending on the requirement and the lower level functions will be identified. And most of the time this lower level functions are the configuration item which will be identified in the generic architecture. And from this functional architecture we convert that in to a generic architecture this actually shows the functional architecture which actually we saw in the previous lectures also.

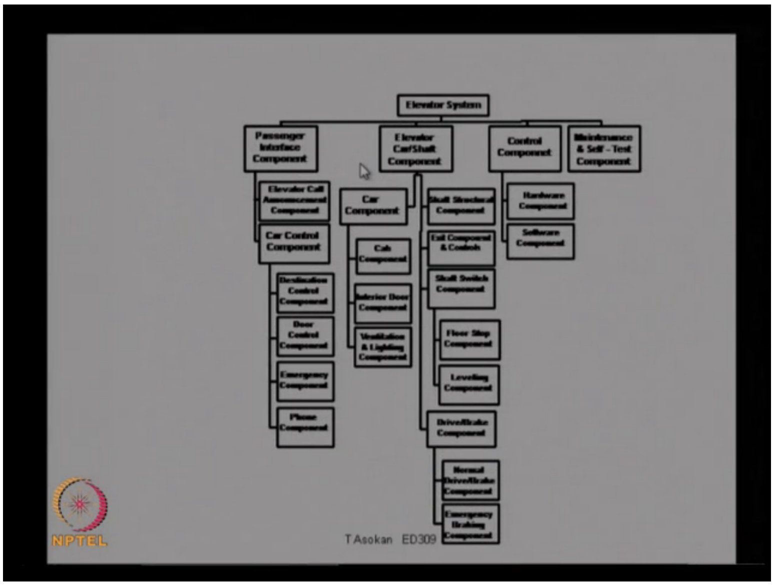
So, this is the from this functional architecture we develop the next level of physical architecture.

(Refer Slide Time: 17:06)



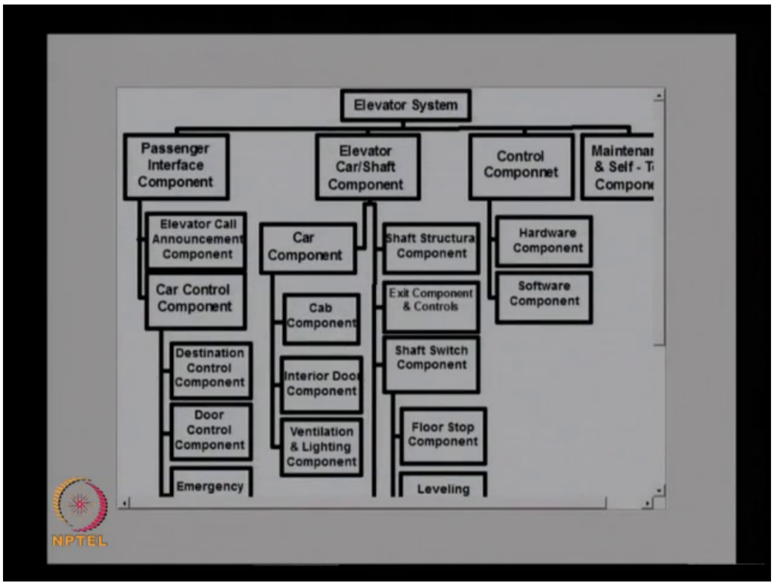
So, as an example I can show you this is the functional architecture for an elevator system. And we saw that this is the top-level function where move passengers between floors and then accept passenger requests and provide feedback and control elevator cars these are the sub function using functional decomposition. We identified how to get these functions and from there we can actually decompose them into further. Like, provide input output interface process data provide control commands etcetera. So, this way we can actually develop.

(Refer Slide Time: 17:33)



Now, when we convert this into a physical architecture, then we can see that the physical architecture can actually be or a generic physical architecture will be more like direct conversion of the functions into generic components.

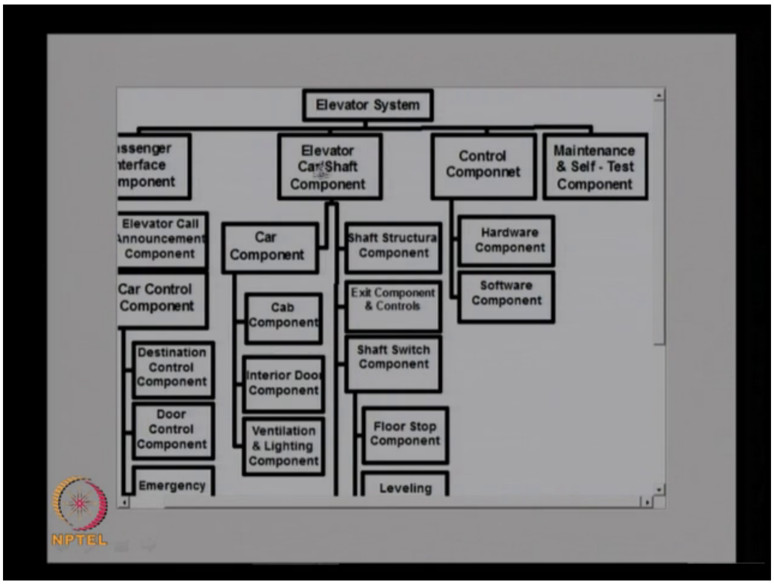
(Refer Slide Time: 17:47)



So, here you can see that the top-level component in terms of the generic architecture is the elevator system. So, provide elevator services is the function, and elevator system becomes the generic component which will provide this one.

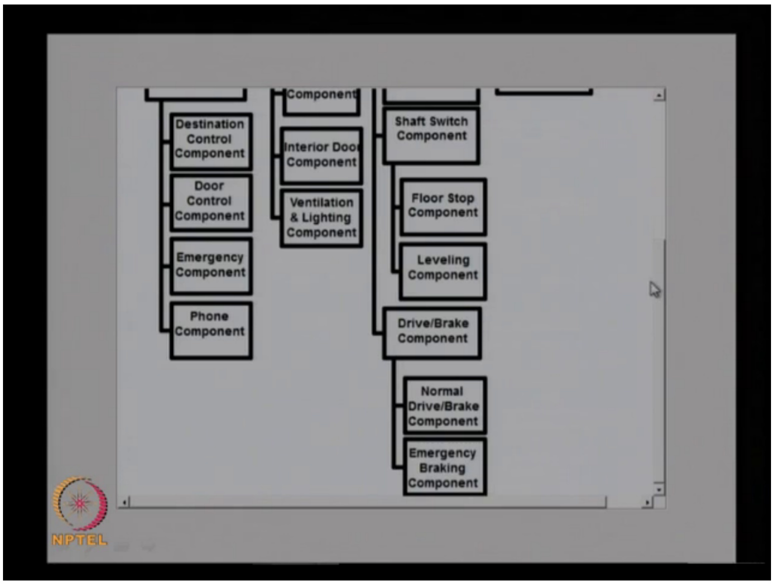
Similarly, provide input output interface for the passengers is the one of the functions. And here the passenger interface component becomes the generic elements. Similarly, elevator car or shaft component becomes the generic elements for the next function. Similarly, control elevator cars the control component becomes the generic element and maintenance and self-test component becomes the generic elements for maintenance functions.

(Refer Slide Time: 18:31)



So, again we are not specified any specific control component or the elevator car or the type of interface here. So, we have just converted the function to a generic physical element. And same way we can actually divide this component or the these assembly to sub component like elevator call announcement component, car control component and destination control door control emergency control, then car component these a cab component interior door component ventilation and lighting component. Like this shaft structure components exit component controls shaft switch component floor stop component leveling component drive break then here itself we can again decomposed into normal drive emergency breaking like this.

(Refer Slide Time: 19:13)

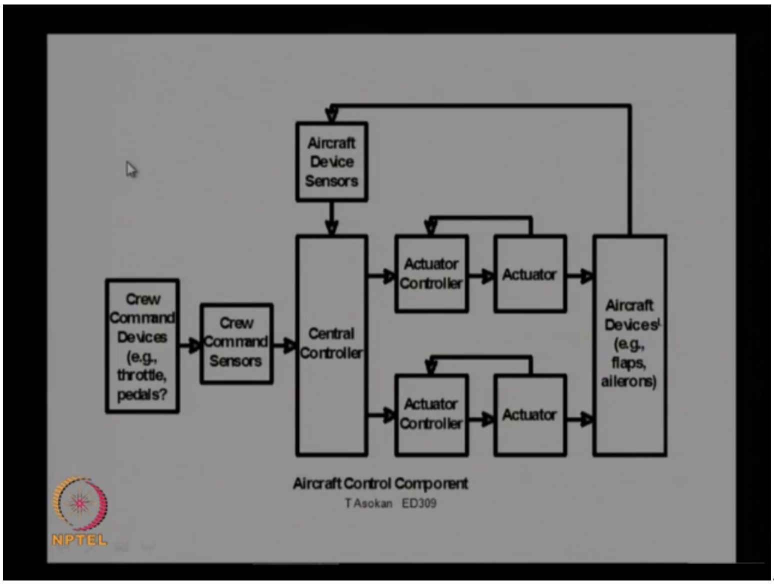


So, what we are trying to do here is to convert the functions or the functional block into the generic names or generic components. So, the here without any specification of the car component or the control component or the breaking component or ventilation component, we are simply writing down this as a generic name like interior door component or the ventilation and lighting component. So, what should be this particular element what should be the specification of this will be decided in the next level.

So, the first level is basically identifying the generic names or generic components and then writing down them in the functional in the same as functional hierarchy or the similar way of functional hierarchy, and getting the generic physical architecture. So, this is the way how we get the generic physical architecture. So, starting with the functional hierarchy, we convert the functional elements in the functional hierarchy to generic elements and write down them in the hierarchical way then you are getting the generic physical architecture.

So, from these onwards we will be going to the next level, where we try to identify the component for each of these elements. So, here we are not specified the elements. So, next level we will try to identify the elements for these components.

(Refer Slide Time: 20:33)

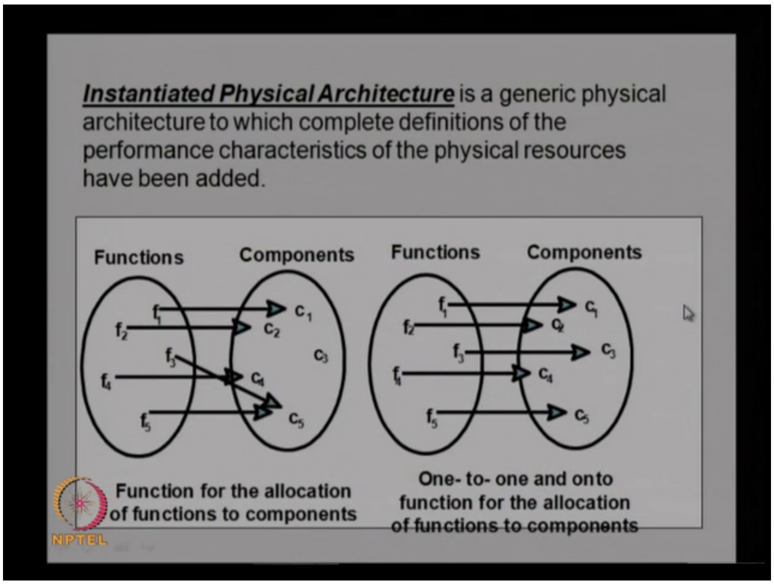


So, these another way of representing the hierarchy generic architecture, here again this is for the aircraft component here. Again, we do not specify the particular elements here, but we will give a generic name for this like crew command devices, crew command sensors, central controller aircraft device sensors, actuator controller, then actuator, then aircraft devisers, actuator controller and actuator.

So, this is the way we write down the generic physical architecture. We do not specify what kind of a sensor we are using, or what kind of an actuator we are using and what kind of controller we are using we simply specify the generic name of the those elements. And from based on this then will go to the level where we can identify few actuators. And then choose one of the actuator for these replace this with the actual actuator we are using or the actual component, then we are getting the next architecture which is the instantiated architecture.

So, in order to get the components names we need to do a brain storming and then try to find out what are the possible option for these elements. And this is done through the this is actually the same figure previous figure, explain in detail here or more for better clarity. So, I can see the control the generic names of this elements written over here.

(Refer Slide Time: 21:55)

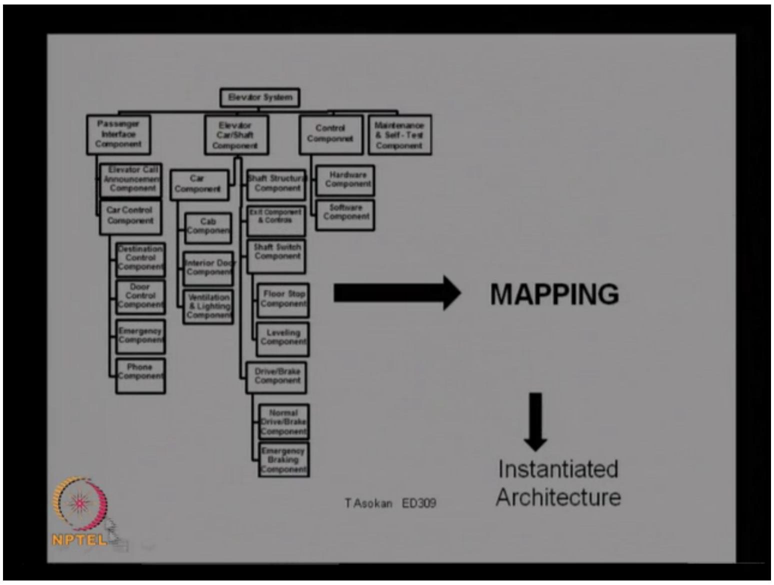


So, to get the instantiated physical architecture, which actually is a architecture to which complete definitions of the performance characteristics of the physical resources have been added. So, we will take the physical architecture or the generic physical architecture, and then we add the names of the components to these elements, and then the that gives us the an instantiated physical architecture. So, this can be done by mapping the functions to components, and then doing the and checking where the components actually satisfy these functions.

So, we can do different kinds of mapping of functions, one is known as the one to one mapping or on to mapping. There are different ways of mapping, this one we will see detail and the and we allocate the architecture. So, in this for the time being you can understand that. This is basically used to identify whether all the functions are being map to components, or how these components are map to the functions whether there is a one to one or on to mapping. Or there are functions which actually satisfy multiple components which actually satisfy multiple functions, or there are some function which have been left out without any component.

So, this things can be identified using this kind of mapping functions. So, instantiated architecture what we try to do is to identify the components for the physical elements. And then write down this elements with the architecture or the generic physical architecture. So, when we add the actual physical component to the generic physical architecture.

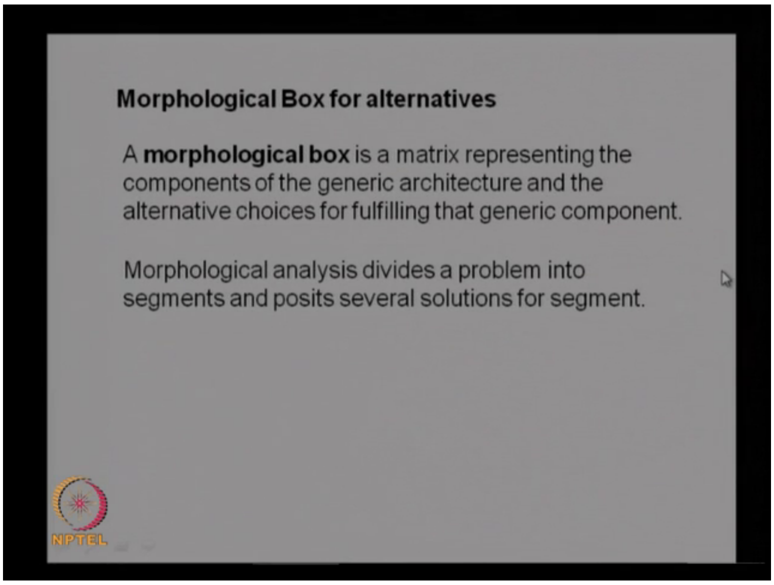
(Refer Slide Time: 23:31)



We are getting the instantiated the physical architecture. So, to do this we need to go for the morphological box. Basically, we take the physical architecture, we do a mapping and then we get the instantiated architecture.

So, this mapping in order to do the mapping we need to generate the alternatives. So, we generate the alternatives through the mechanism called morphological box. So, we use a morphological box to get the components, and then we do a mapping of this component with the generic elements, and then we get the instantiated physical architecture. We will see how to get the mapping.

(Refer Slide Time: 24:00)

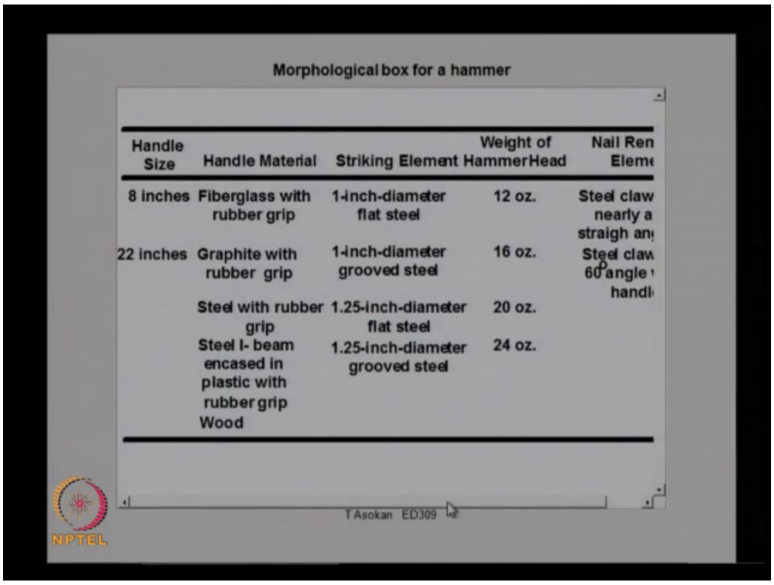


Basically, we will develop the morphological box initially. So, morphological box will be developed to get the alternatives, and then they will be mapped.

As you can see a morphological box is a matrix representing the components of the generic architecture, and the alternative choices for fulfilling that generic component. So, this is the basically a matrix of alternatives for generic components. In the generic physical architecture, we develop the elements generic elements, and then based on these generic elements we can identify all the alternatives possible, and write down them in a matrix format we are getting the morphological box.

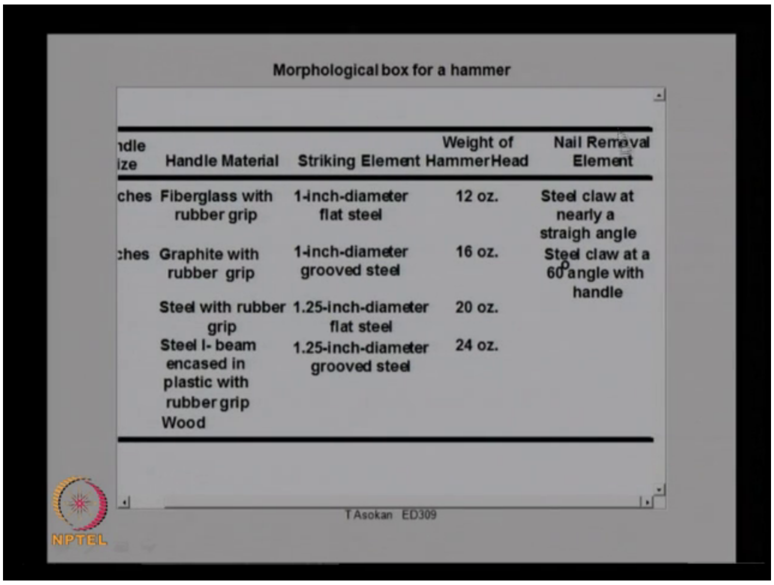
So, this actually divides a problem into segments and posits several solutions for a segment. So, you can actually divide the whole problem into small segments, and then each segments can be unless separately and we can get alternatives for each of these segment using morphological box. So, this an example for a morphological box.

(Refer Slide Time: 24:55)



If you look at the how morphological box are developed. So, this is example for a very simple product called a hammer. So, you should take a hammer as a product, and then how do we develop the morphological box for these products. We look at the generic components and the generic requirements like handle. So, if you take the handle as a product then we can see that handle size and handle material, then striking elements weight of hammerhead, and nail removal element as the generic components in this case.

(Refer Slide Time: 25:24)

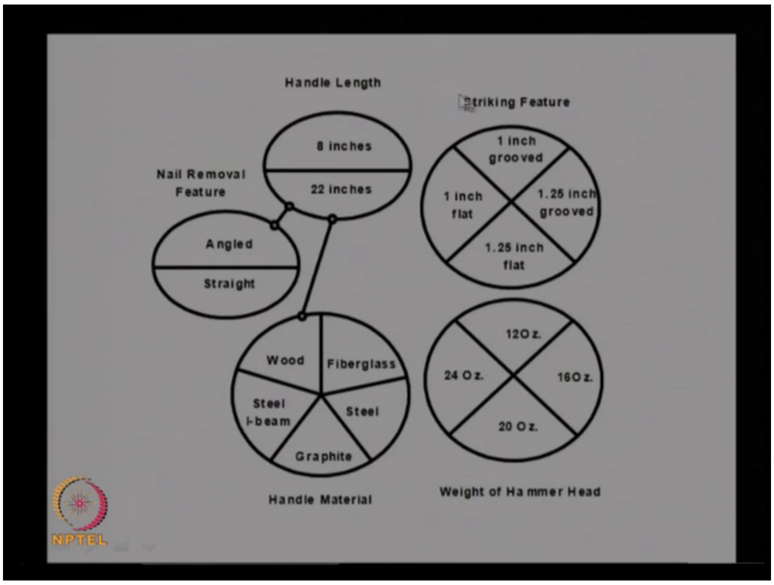


So, these are the generic elements and we need to get develop the alternatives for these generic elements. So, we can actually write down the possibilities here. So, alternatives are we can have a handle size of 8 inches or 22 inches, and we can have different handle materials, fiberglass with rubber grip graphite with rubber grip or steel with rubber grip steel I beam encased in plastic with rubber grip or wood. So, these are the possible options for the handle material.

So, this is the alternatives for handle material and for striking element. We can have alternatives like a one-inch diameter flat steel one-inch diameter grooved steel 1.25-inch diameter flat steel. Or a 1.25-inch diameter grooved steel. Similarly, if the weight of the hammerhead can be 12, 16, 20 or 24, and nail removal element could be a steel claw at nearly a straight angle steel claw at 60-degree angle with handle. So, these are the alternatives possible for a particular system and particular product.

So, this is this is just to explain the morphological box, actual system design will be having much more complex alternatives, much more complex structure because we will be having many generic elements. And then we need to have many options, and here this is a simple product that is why you to see only few options given here. But even with these few options or few alternatives, it becomes very difficult to choose a particular product because particular configuration, because we you need to look at the alternative options here and then the implications of them in actually selecting the alternatives.

(Refer Slide Time: 27:03)



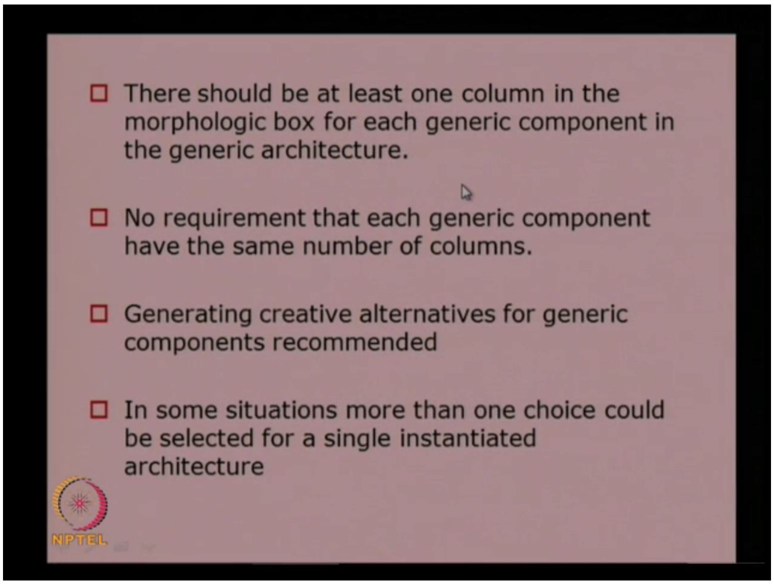
So, here you can see these are the options from the architecture. If you look at the morphological box, you can see that there are 2 options over here, there are 2 options here, and you have 5 options in handle material, and another 4 options in hammerhead and then the striking feature again 4 options. So, how do we actually if you want to make one hammer. So, what should be the possible combinations here.

So, that actually to be chosen by the design based on the requirement, and the customer expectations as well as the customer needs identified. And the initial concept based on that they need to choose the a particular combination from here. But are there are some combination which are not possible, and such combination will be added with the line or connected with the line. So, in that case we will actually eliminate. So, those combinations because of some reasons this 22 inches in an angle combination do not work. Similarly, this 22 inches and wood would not work because of the again implication the design implications.

So, such situations can be easily marked and then we can eliminate them, and then the remaining can be considered for choosing the alternatives. So, this is how we use the morphological box, or we develop the morphological box for a generic element, or multiple generic elements. And then develop the alternatives. This is actually shows in a with more clarity as you can see there is a these are the different options for handle nail removal feature, then striking feature.

Similarly, the weight of hammer head you can create this kind of graphical representation of also from the morphological box and then develop or choose the alternatives depending on the requirements.

(Refer Slide Time: 28:55)



But some of the points to be noted or to be taken care of while developing the morphological box is that there should be at least one column in the morphological box for each generic component in the generic architecture.

So, we have many a generic components, and at least one column in the morphological box for each generic component should be there. So, then only we can actually develop the alternatives. So, for every generic component we should identify the alternatives. So, the meaning of this particular statement is that every generic component should be having many alternatives, or at least few alternatives should be identified for generic components. And there is no requirement that each generic component have the same number of columns. So, it not necessary that there should be same number of alternatives for every generic component, we can have in some cases we can have 2 alternatives or some cases you can have multiple alternatives or 5 or 6 alternatives can be; because depending on the generic component we will be able to identify 2 or more alternatives.

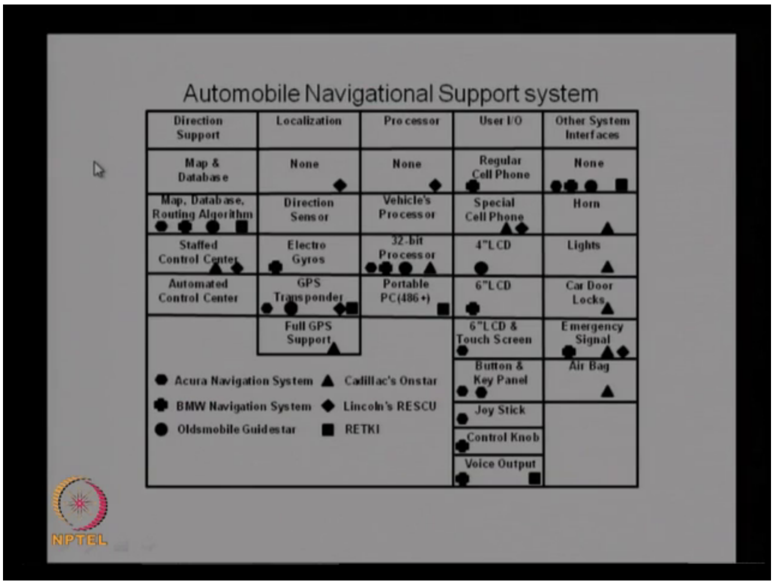
So, there is no rule that then should be equal number of alternatives in the case of generic components. Then generating creative alternatives for generic component recommended. So, always it is ah recommended to have innovative alternatives instead of going for the known and already existing. So, it should be always advisable to go for creative solutions. So, instead of choosing a readymade or readily available component, if you are really interested in making a product or a system which is different from the existing one and you want to make it little bit more innovative, then it is advisable to go for creative solutions for alternatives.

So, this actually helps us to identify the possible options and the to apply our creativity to get the alternatives, and make the system much more simple and innovative compared to the existing system. So, always try to go for creative solutions for alternatives. And in some situations, more than one choice could be selected for a single instantiated architecture. So, this is again think that when you have multiple choices. It is not compulsory that you choose only one choice for that particular architecture.

There is a possibility that you can have multiple choices for a particular generic element, and then add this as possible options so that at a later stage if you want to add more features or you want to increase the features in the product they can be easily added to that one. For example, if you want to have some peripheral devices for a computer. So, we can actually have multiple options there which one to be added can be decided at a later stage. Or we can actually have options for adding this at a later stage. Therefore, we can have peripheral devices as a generic element, you can have a speakers then USB's or printers, scanners, and camera; those can be added as alternatives in the instantiated physical architecture so that this actually becomes part of the instantiated architecture instead of choosing a single product or a single element as from the morphological box.

So, and some situations you may like to have more choices in the instantiated physical architecture.

(Refer Slide Time: 32:20)



So, let us look at the one example where how do we develop this instantiated architecture basically, from the morphological box. This actually shows an automobile navigation support system, this actually gives a better clarity you can see here there are many generic elements here. This is basically a system for automobile navigation support suppose you have a car and you have a navigation system in the car, and what are the generic elements in that navigation system, and then how do we convert this elements into physical elements will be the question here. And to do that will develop the morphological box.

So, here you can see direction support is one generic element. Localization is a another generic element, and then processor, then user input output and other interfaces. So, this is are the generic elements in the architecture or in the generic physical architecture will identify these elements without specifying the actual product. Then we can see that what are the options for direction supports. Here this is the morphological box for the navigation support. So, for direction support you can see, you can use a map and a database is a very simple you can use map and database, or you can use map database in the routing algorithm. Along with the map and database we can have a routing algorithm, and then we can have a shaffed control center where the navigation support can be directly provided to the driver or you can have a automated control center.

So, these are the options for direction support. So, in order to support the driver for the directions, we can use any one of these options. And what this symbol shows here are the possible manufacturers or commercially available system. So, here you can see that this was the acura and navigation system, then BMW or oldsmobile guidestar, then cadillacs or lincolns and RETKI. These are the commercially available in the market. So, this actually shows that for this particular one you can use any one of these. So, we are more like specifying the details of that particular component in terms of the commercially available product.

So, once you say that out of these 4 anyone can be chosen. So, indirectly I am making the specification of this particular component. So, from the generic name we are going over to the actual physical element here similar this staffed control center these 2 operators who provide the staffed control center. So, any one of these can be used for this particular application.

So, for this a generic element of direction support, we have many options we have identified 5 options, and we identified few real physical components also. So, any one of these can be selected. Similarly, for localization we can decide not to have a localization, or we can have a direction sensor for localization, and you can have electro gyros or GPS transponder or a full GPS support. So, in order to get the localization, if you do not want localization, you can actually decide the driver can himself decide where he is or he can identify the location. Or we can have some direction sensors to look at the directions and then decide where the location is, or you can have gyroscopes or GPS transponder and full GPS support.

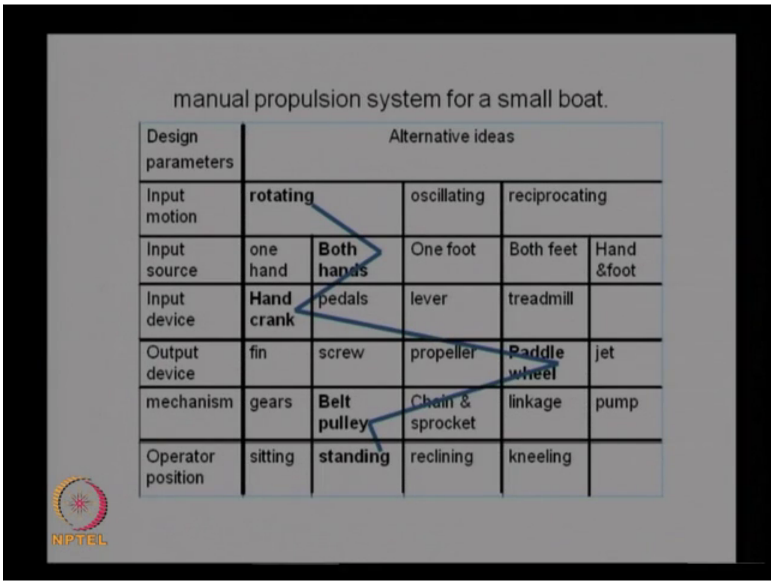
So, any one of these methods can be used for localization of the car. And then again, we are specifying it is available commercially giving the choice is also here. So, any one of this can be decided or chosen for this suppose the decides to go for GPS transponder. So, here is 4 options, and out of this he can choose the one which he wants to have, the specification. Once we choose a particular component the specifications are fixed. So, no need to worry about the component details here because it is already existing.

And the same way we can go for other generic elements like processor can have a processor in the vehicle and the vehicle processor can be used. Or a separate 32-bit processor can be used. Or a portable pc can be used. So, any of these options are possible for a processor and the user input outputs you can have a regular cell phone or a special phone for this or a dedicated phone for this or you can have a display LCD display for in (Refer Time: 36:32) or a display with the touch screen or a button and a key panel joystick control knob voice output.

So, there are multiple options for the user input output interface. And the last one basically you have other interfaces, you can have some interfaces with the system or you can decide not to have any interface with the system. Basically, other system interfaces or you can have a interface with the horn or light. Or we can have a interface with car door locks emergency signal and the air bag. This is basically to help the customer or the driver to contact the services emergency service provider. So, you can actually link the passenger navigation support system with the an emergency system.

So, that this becomes an additional interface. Again, there are multiple options. You can actually connect with the horn lights or car door or emergency signal. So, you can see here we identified few generic components. And for this generic components develop the alternatives. And as I mentioned it not necessary that are the generic components they have equal number of alternatives. Here you can see. There are variations in the alternatives. And based on these alternatives we can decide what actually should be the configuration, the physical configuration of the system and once we select these components. And then write down the physical architecture will get the instantiated physical architecture. So, we are identifying the architecture with the components and the component specifications. So, you are getting instantiated physical architecture.

(Refer Slide Time: 38:15)

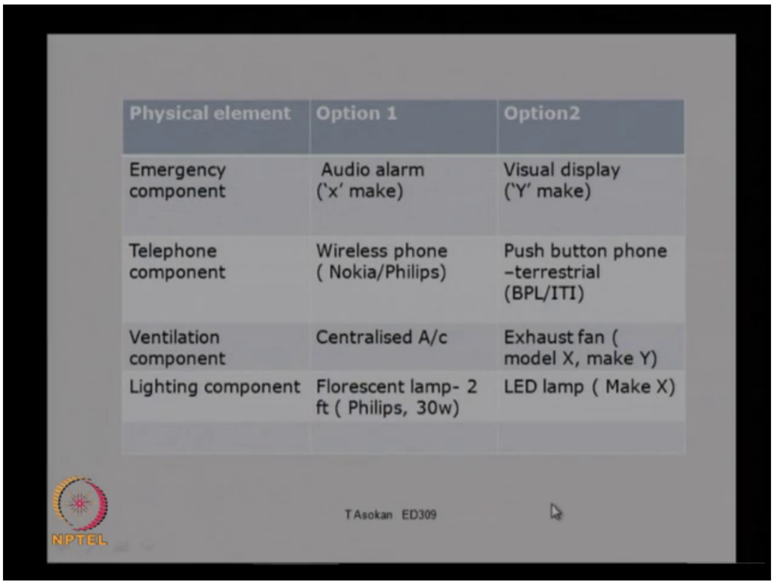


So, these again shows another example for morphological box, and then how do we choose a particular configuration for it. So, here you can see it is a manual propulsion system for a small boat. Again, it is very a simple example to show how to use the morphological box. So, here you can see these are the design parameters or the generic elements input motion input source input device, output device, mechanism and operator position. So, these are the requirements of generic elements. So, here you can see the input motion can be rotating one or a oscillating one or reciprocating another option. And then input source where be both hands, then input device, hand crank or pedals output device, fin jet or propeller mechanisms are gears belts pulleys linkage pump etcetera. And operator position is sitting standing reclining or kneeling.

So, these are the alternatives for this particular feature or the generic element. And based on this generic element we identify the alternatives, and then from there we can actually choose a particular configuration. So, in this case you can have that by manual propulsion system could be rotating a type using both hands with the hand crank, and a paddle wheel with the belt pulley and in a standing position. So, we can have a propulsion system were operators is in a standing position, and it is connect through a belt pulley to a paddle wheel and using hand crank both the hands it is being rotated and getting the manual propulsion system.

So, this is how we use the morphological box for choosing a particular configuration. So, the same principle can be applied in the case of system design also. We will identify are the alternatives, and then choose particular based on the requirement we choose one of these alternatives or in some cases multiple alternatives to choose our physical structure. So, this is how we develop the physical architecture. These actually explain the same figure previous figure in with the better clarity. So, you can see here all the options are listed here, and the choices are marked with line or here.

(Refer Slide Time: 40:28)



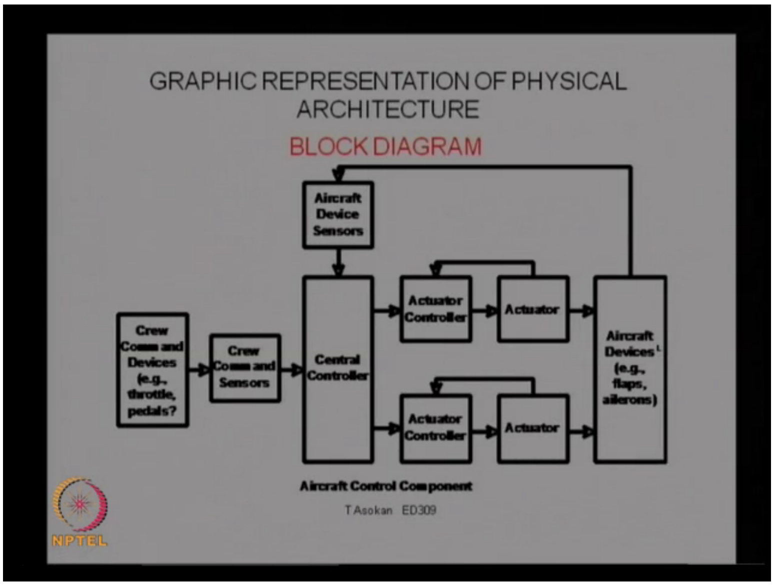
Physical element	Option 1	Option2
Emergency component	Audio alarm ('x' make)	Visual display ('Y' make)
Telephone component	Wireless phone (Nokia/Philips)	Push button phone –terrestrial (BPL/ITI)
Ventilation component	Centralised A/c	Exhaust fan (model X, make Y)
Lighting component	Florescent lamp- 2 ft (Philips, 30w)	LED lamp (Make X)

So, if you look at the elevator case study previously we can identify few elements. So, the physical elements I just shown only 2 options here of course, you can have multiple options as an example I have shown here. So, we identified that in the generic physical architecture there are an emergency components, and then there is a telephone component a ventilation component and lighting component. So, we identified these are the generic components. Now how do we actually choose a alternative? Or how do we get the options and then instantiated the architecture? So, will write down the emergency component is audio alarm. You can give the makeover here or you can have a visual display another make, or the telephone you can have wireless phone or a push button kind of phone terrestrial or a mobi this as a mobile phone, or a terrestrial phone with a push button. We can we can again define the manufacturer and the make.

So, it will be very specific to that particular product. And you can the ventilation component we can have a centralized air conditioning for all the cars or you can have a simple exhaust fan connected to the elevator room. Then again you can specify the model or what are the size specification can be given clearly here. Then you have the lighting component you can have florescent lamp, or a led lamp. And based on the requirement we can choose any one of these, and then write down the component name the in the generic physical architecture if you write down the audio alarm as an emergency component then it becomes an instantiated architecture.

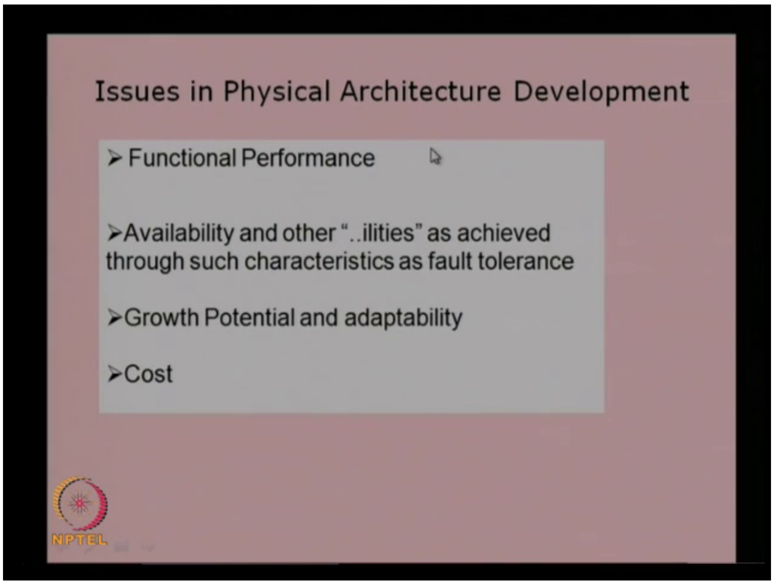
So, like this if you write down the component name and the specification for the each and every generic component, then the whole architecture becomes the instantiated physical architecture.

(Refer Slide Time: 42:06)



And there are the different ways of representing the architecture. So, this is the generic structure and in terms of a block diagram. So, we have the hierarchical way of representation as well as the block diagram way of representation. So, hierarchical way we already saw in the functional architecture, how it is done in the block diagram we write down the generic architecture, and then we write down the name of the component adjacent to this then it becomes a instantiated physical architecture block diagram way of representing.

(Refer Slide Time: 42:38)



That is the way how we develop the instantiated physical architecture. And when we develop this architecture there are different issues to be taken care of these basically the functional performance.

So, the we make sure that the physical architecture, we do a proper mapping of the components to the functions. So, the functional performance need to be taken care of and then made sure that we identify the required functions and the components needed for this. And then other things like the availability and other utilities and the reliabilities all those things to be taken care of and then we need to provide the component for those functions also. And then the future growth potential and adaptability to be taken care of. We need to ensure that there is sufficient potential for the growth, and then the cost of the whole pro product also to be taken care of.

So, in next lecture basically we will try to see, how to take care of this features basically the reliability of the system or the fault tolerance of the system. And how do we provide the components physical components to give these functionalities in the system. The previous structure we looked at the from the function architecture we develop the generic physical architecture, and then from generic physical architecture we try to develop the alternatives and then instantiated physical architecture. In order to complete the physical architecture, we need to add the other features like fault tolerance and other aspects of the system. And if you add the components or we identify the alternatives for such functionalities, and we then and add these component to the system then only be the complete architecture will be developed.

So, in the next lecture we will look at the possibilities of adding the fault tolerance and the alternatives for fault tolerant functions, and identify the components for fault tolerant functions and add these components into the physical architecture in order to complete the instantiated physical architecture. So, we will discuss all these points in the next class.

Till then good bye.