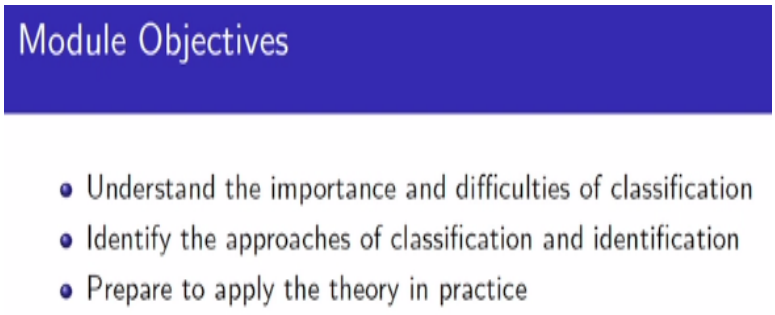
Object-Oriented Analysis and Design
Prof. Partha Pratim Das
Department of Computer Science and Engineering
Indian Institute of Technology – Kharagpur

Lecture – 28
How to Identify Classes and Objects

Welcome to module 16 of object-oriented analysis and design. We have so far been taking different analytic looks into complex systems, their attributes and variety of commonalities and approaches that could be adopted to analyse these complex systems from various aspects of objects and classes and their relationships.

In specific, we have looked at different elements of major and minor elements of objects; we have looked at the nature of objects and that of classes we have looked at their relationship in depth. From this point onward, we are slowing taking the course towards practicing all the different principles that we have already discussed.

So from this module onwards where we will talk about some of the major tools and techniques of actually identifying objects and classes from a given specification of written, verbal, interactive or pictorial kind and start taking example and active exercise example with a leave management system and some of the other systems to illustrate how as a practitioner you actually engage in the object-oriented analysis and design processes.
(Refer Slide Time: 02:05)

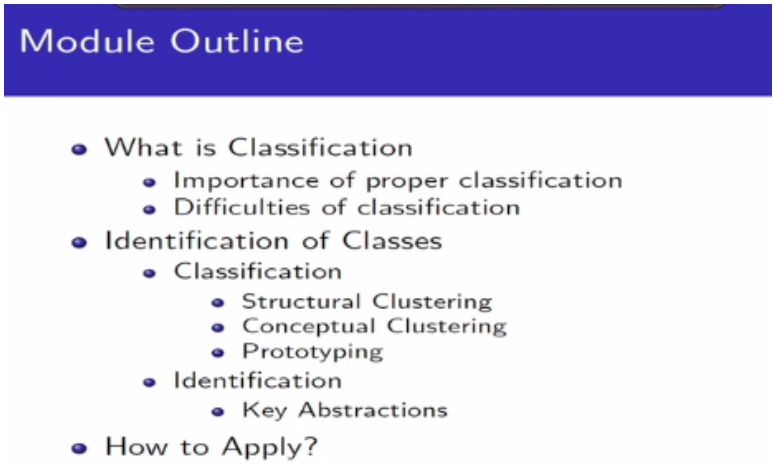


So this module is to understand the importance and difficulties of classification. We have

seen in the earlier discussion in theory that whatever where we look at the system, it is very critical to identify objects, identify classes. So given the whole bunch of concepts and dynamics of the system, it is a critical requirement to classify them according to certain commonalities which we can capture in terms of the classes.

And they have been a lot of work on this in terms of classical approaches and then they were followed by several modern approaches. We will not go into that historical development; we would rather focus very specifically into how we can adopt these and apply them in practice.

(Refer Slide Time: 02:54)



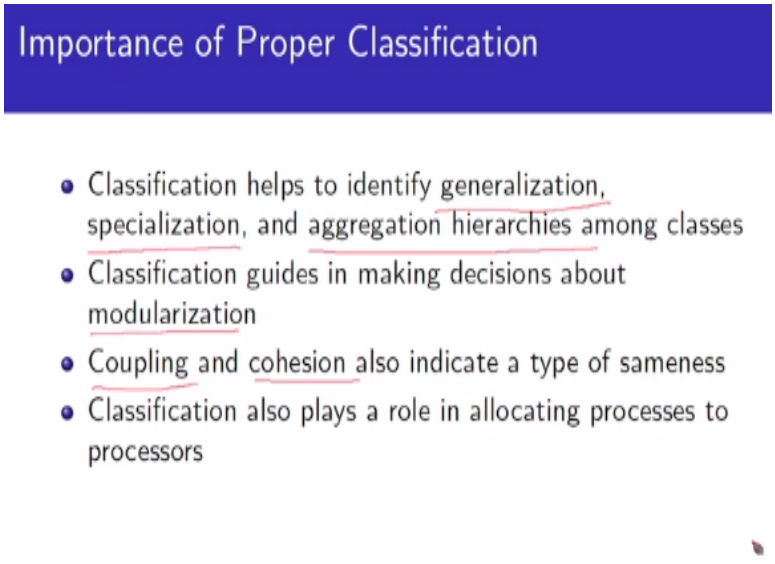
This will be the gross outline of the module and we will continue to display this on the left hand of every screen.

(Refer Slide Time: 03:07)

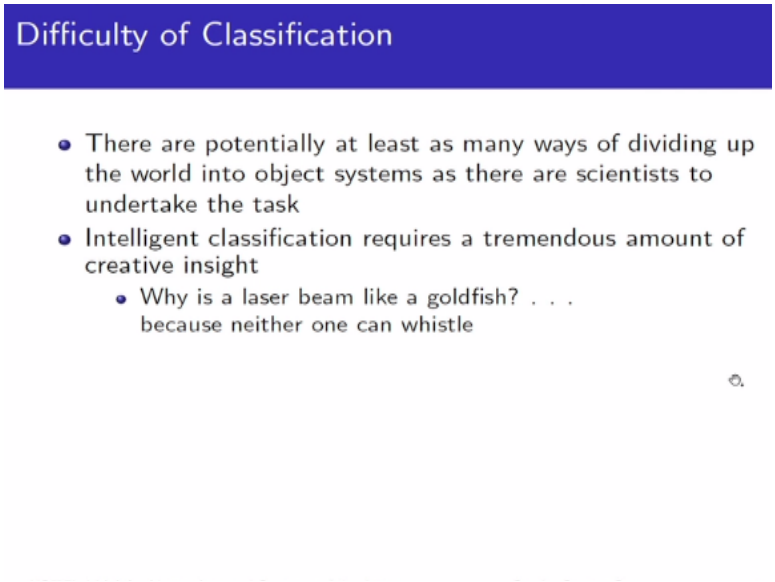
So, first let me briefly talk about what is classification, of course, most of you would know that classification is a process whereby we put some kind of an order in other knowledge. Some kind of a boundary around things that are similar, tasks that are similar or actions that are similar or they share something.

So here in this illustrative diagram, I am trying to show that we often, when we these days go to dispose garbage, then we get a number of different vats, some written with glass, some written with plastic and so on and depending on what we have whether we have a vegetable waste, a biological waste or we have a plastic cup in depending on that, we need to decide which vat we should put our garbage into.

So this process that we are doing mentally is actually a process of classification and in general we will need to do this for given the system that we would like to analyse.
(Refer Slide Time: 04:17)

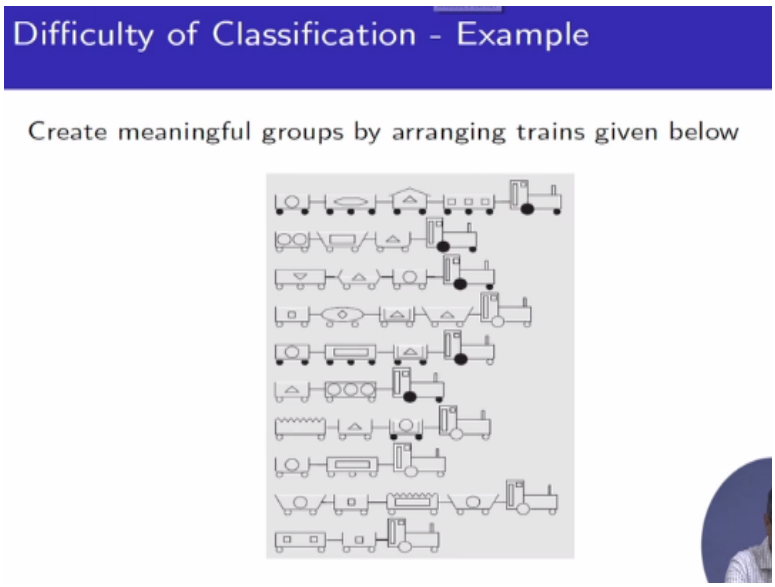


So classification will help to identify generalization, specialization and aggregation amongst the hierarchies amongst the classes. It will also help in making the decisions about to modularization that is what all items, what all classes, what all processes we should put together. In terms of coupling and cohesion, the two main quality matrix that we have studied in the earlier module indicating the sameness is also decided primarily by the classification and it plays a critical role in terms of allocating processes to different processors.
(Refer Slide Time: 05:10)



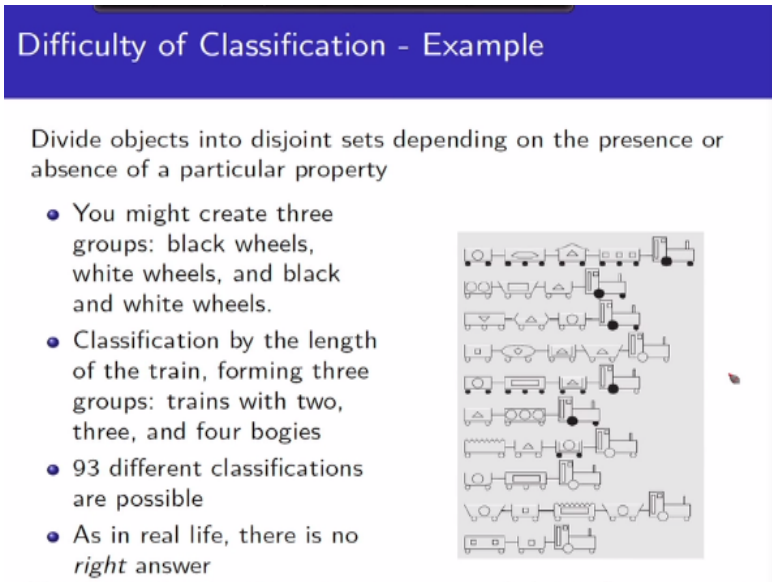
So having said that, let us recognize that the classification is not an easy problem because given the same scenario multiple observers could classify them in multiple different ways because it is not, there cannot be any very straight forward or mathematical or algorithmic ways to classify given a scenario because it pretty much depends on the experience of the observe where it depends on the domain, it depends on what you and I identify as the most discriminating parameter in the classification process.

(Refer Slide Time: 05:53)



So it is a, it is quite a difficult process to go with. So just to illustrate with a little bit more concrete example, let us take this example from (()) (06:04) book where we are trying to create, we have given a pictures of ten different trains right. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 different trains and these trains have an engine and a couple of compartments that it drags and we would like to create meaningful groups, meaningful classification of these trains.

(Refer Slide Time: 06:35)



So, we can divide them, classify them according to, if you just do not have a guideline, if you just look into this picture and you will be able to create different kind of classification depending on, for example, the presence and absence of a particular property, you could decide that okay. I will look for certain common properties. This is pretty much like the you know the IQ test stuff that we keep on doing all that time.

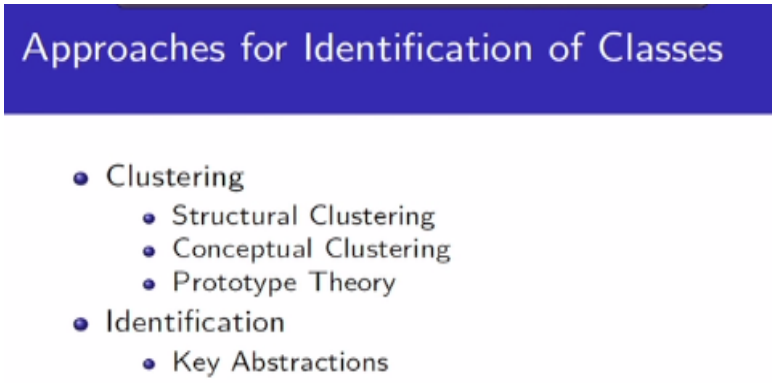
For example, we can observe that different trains have different kind of coloured wheels. Some have black, some have white, some have white and we can decide a classification based on the colour of the wheel. So we will have the trains which have black wheels, white wheels and which have black and white wheels both. So if you look at that, this is completely black wheel but this is completely white wheel, this is completely white wheel whereas this one is black and white wheel. So we can classify the trains according to this.

Alternately, we could say no, it is, that is not the right way to classify, I would really need to understand how long the train is, so I will classify based on the property of the length of the train. So if you look at that then they will fall in three groups where trains with which have two bogies, three bogies and four bogies, like this has two bogies, this has two bogies whereas this train has three bogies, this train has four bogies and so on.

Accordingly, we can classify into different groups that will get three classes for that and in this way if you try to look at all this different patterns that exist in this image then as it turns out that there are 93 different classifications that are possible and in real life unfortunately,

and that is where the major challenge of the practitioner lies is there is no exactly right answer where this is right and this is absolutely true, there is nothing like that. So we will have to work within this level of uncertainty and ambiguity that exists in classification.

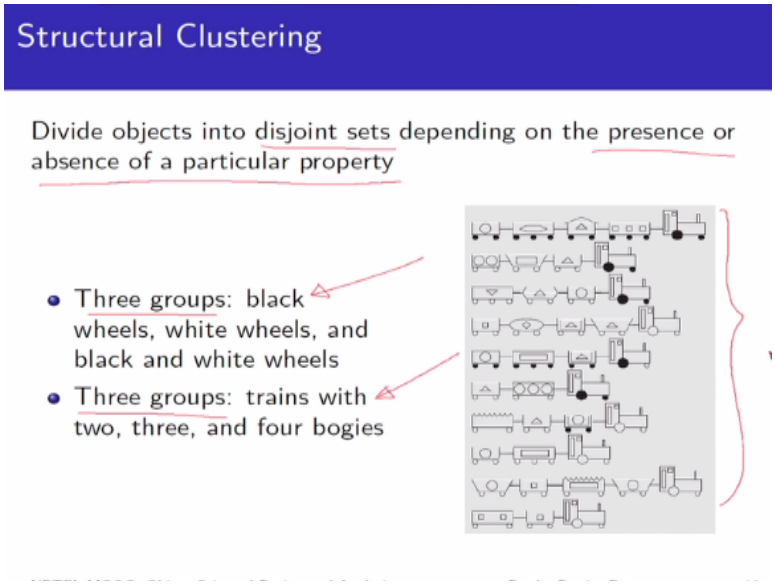
(Refer Slide Time: 08:50)



So based on this, the identification of classes have been tried in various different ways and as of the current practice it primarily relies on one on clustering and then next is on identification. The clustering is the process by which that you say that given all the different concepts that we have in the domain how can we group them together, how can we classify and make groups. Every group that you make by classification we say is a cluster.

And the other is once we have created clusters, then we identify what is the characteristics of that cluster, so that is the two basic process and once we have been able to identify the characteristics, we say that these are the key abstractions and as we had mentioned several times before, key abstractions becomes classes in your system. So in terms of clustering there are three main approaches which we will slowly illustrate in the next couple of slides.

(Refer Slide Time: 09:52)



We start with the structural clustering. In the structural clustering the objects are divided into disjoint sets depending on the presence or absence of a particular property, pretty much the way we were addressing this earlier. So if we look at the colour of the wheels, then we get three groups that could be one structural clustering. We will again get three groups if you look into the number of bogies that every train has and so on.

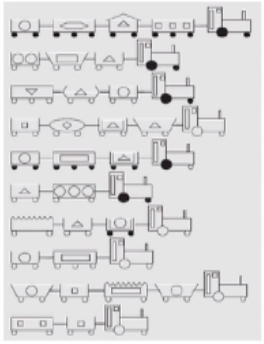
So we are using different properties like colour of the wheel or the number of bogies and to cluster the given set of possible objects into different cluster groups and this process, since it is based on certain concrete properties is known as structural clustering. So in many situations, a structural clustering, it is relatively easy to do structural clustering because once you can (()) (10:47) on some property, then you can apply it or apply the property with its all possible values and they lead to the different clusters that will emerge from the system.

(Refer Slide Time: 10:58)

Conceptual Clustering

Attempts to explain how knowledge is represented

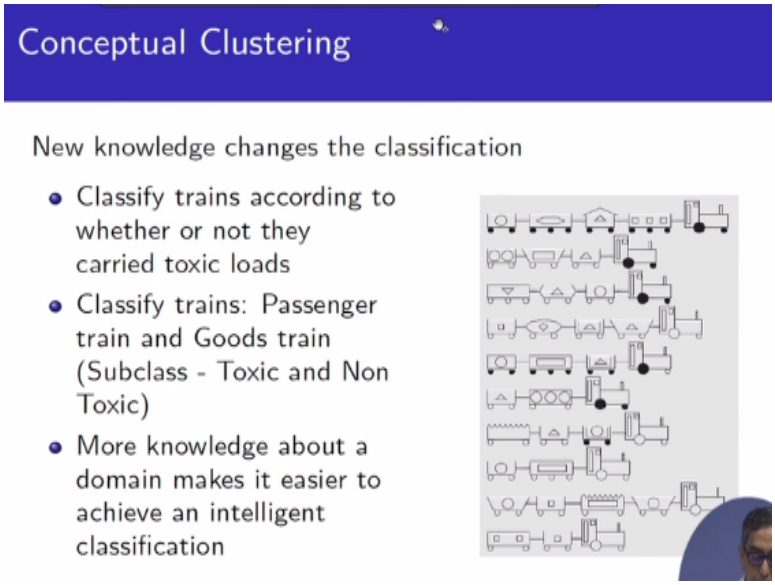
- Let us change the requirements as in real life
- Suppose
  - circles represent toxic chemicals
  - rectangles represent lumber, and
  - all other shapes of loads represent passengers
- Try classifying the trains again



The next is often okay, I can change the game as is the reality in most cases that it is not only one or two structural parameters that define the way objects will work, the way objects will interact. So let us look at some of the conceptual properties. It is like, let us say, you can see in this diagram in this picture, that the bogies have different kinds of symbol like this as a circle symbol, this has a rectangle symbol, this has a triangle symbol, this has another circle symbol and so on.

So let us, suppose we know that these symbols characterize the type of content that the bogie transports. So let us say, just hypothetically, the circles represent toxic chemicals. Let us say rectangles represent timber logs, the big wooden logs and may be all other shapes represent passengers. So if we say that this is our basis of classification that we will classify based on these concepts then let us see where does it take us.

(Refer Slide Time: 12:18)

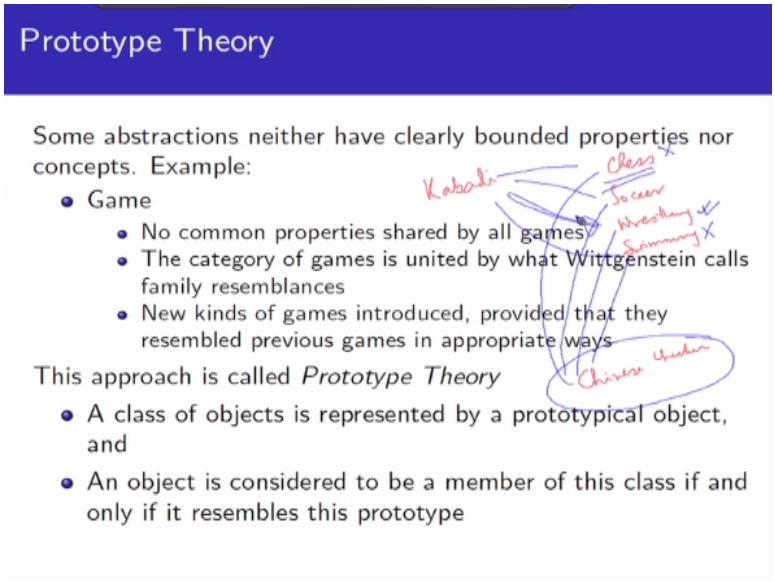


So we can classify the trains according to, for example, whether or not they carry toxic goods. We can classify the trains based on whether they are passenger trains or goods train, that is whether they just carry passenger or they just carry goods. Again if we do based on this, then we can sub classify that, is it a passenger train which carries toxic goods or is it a goods train which carries toxic goods.

So as we get more and more knowledge, now if you just look at the picture, if you just look at the diagram, then the kind of structural clustering that we had been doing is a first attempt that you can make because it is immediately visible. But here what I have done along with that picture I have given you more information from the domain that circles main toxics items, the squares mean lumbers and other means passengers and so on.

So using that you can look at to the objects in a more conceptual framework and perform a better classification in as we have been highlighting. So this process as executed is known as the conceptual clustering process and it is subversive to the structural clustering process.

(Refer Slide Time: 13:35)



A third approach which at times is taken is where you will find that abstractions neither have a clear bounded properties nor a very well defined concept. In the earlier two we had either had clear bounded properties like colour of the wheel like number of bogies or concepts like toxicity, like being a passenger train or a goods train but in some domains, in many times, we will see that the abstractions do not offer that.

For example, if you talk about game now, there is no common property which is shared by all the games that we have or that we practice but the interesting thing is that usually if we say that this is, we are talking about Chinese checkers, we are talking about chess, we are talking about soccer, we are talking about wresting. We all would be able to identify that it is a game.

So the classification in terms of games and in terms of varieties of games is not easy using the bounded properties or concepts. So it needs to be identified by what is commonly known as family of resemblance that what you do is a simple approach, you identify, you suppose that a class of objects is represented by a prototype, is okay, this is the prototype that I have.

So I have a prototype which is a just let us say, and then we consider, so let us say that all the possible classes that we have, they have prototypes. So given a particular game, we would like to see what is a prototype that resembles the given one most closely. So let us say for, if we have classes like represented by chess, represented by soccer, represented by wrestling, represented by swimming.

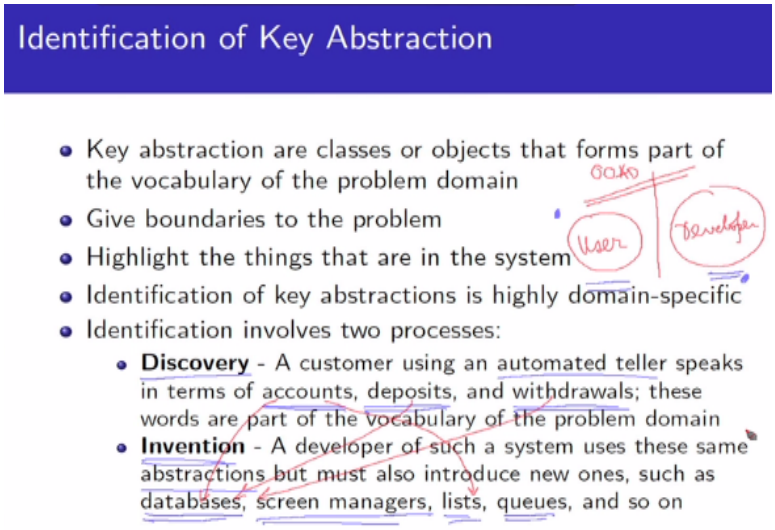
Suppose these are the representative ones then if you are given say Chinese checkers, what

you will try to do, you will try to analyse that what it is more similar to of these different prototypes and of course if you analyse that it is expected that you will come to the conclusion that the Chinese checkers is most similar to chess and we will classify that along with chess.

And then you will try to look for some common property that may have been shared between them for example, you could come up then with an identification that okay, you classified it this way because both of them are board games but suppose if you are given a different game, say if you are given say, let us take kabaddi, So if you given kabaddi and you try to again similarly try to do this measures then certainly you will not find it similar to chess, you will not find it similar to swimming.

Possibly, you will find it most similar to wrestling and you will classify kabaddi to be in the same cluster as of wrestling. So going by this kind of a prototype based approach is known as the prototype theory which is very common in identification of abstractions in classification of abstractions particularly when the structure and the conceptual approaches do not produce any meaningful results.

(Refer Slide Time: 17:36)



Now once you have been able to perform the cluster then you need to, you will get a whole lot of so in the system, now you have various different clusters that have been formed, may be too many of them. Now, your next process is some kind of a filtering that you need to do, is if you say that everything that I could cluster into segregable, identifiable items, if I say that everyone of them has a class and it should have its own relationships and identification

and all that, it might become, you might have too much of incoherent set of system design.

So then, you try to be more going to more refinement and primarily try to design on key abstractions. That is out of these cluster, out of these classes that we have created, what are the key ones that are important and that is a very subjective decision. Primarily, the approach that (()) (18:49) takes is key abstractions try to follow the vocabulary of the problem domain that is what is the predominant vocabulary the problem domain is using and what is the boundary of this problem.

It also has to highlight the things that exist in the system and in this process the whole identification of key abstraction is extremely domain-specific. So again, it is a difficult process but if you focus on some of the typical characteristics of the system that you are discussing what are the abstractions that is classifications as identified by you, what are the abstractions that are more often talked about.

What are the abstractions that are more often active or take part in critical activities. What are the ones which fall within a certain boundary of the system and so on, if you focus on those, it will become usually easy to identify the key abstractions. Now at this point you should also note that identification actually involves two processes, one we say is discovery, other we say is a invention.

Now the basic difference between the discovery and the invention is, if you look in to the whole requirement of OOAD, then we have two sides of this whole, one is the user and one if the developer right. That is the basic thing that we are trying to do right. The user has a, is with a system, user needs certain functionalities to be built on the problem domain, that the user uses and you are developer, we are the developer who is trying to this.

So the user has certain view of the system. So if you ask the user is to what the key abstractions are, user certainly talks about the vocabulary of the problem domain in pure form. So if the user is using an automated teller, then the user will talk about accounts, automated teller means the typical ATM machines or the bank counters where you can withdraw cash.

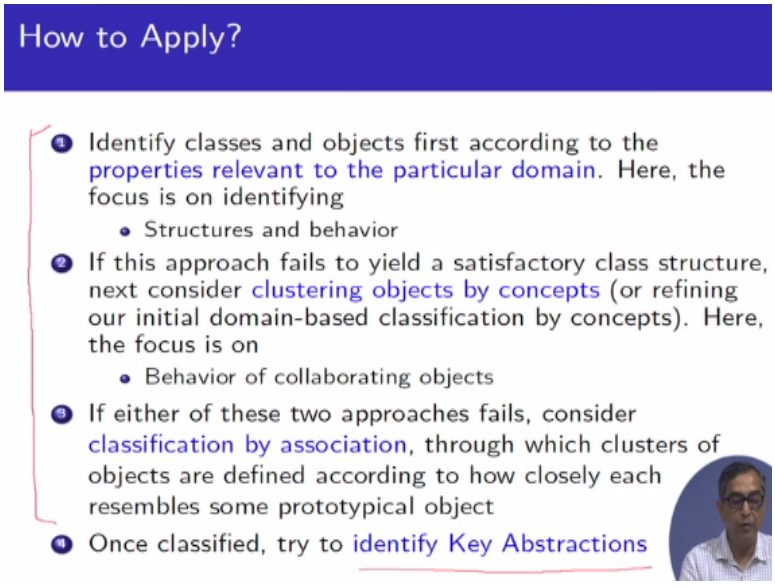
So that kind of user will talk about accounts, deposits, withdrawals, so these are the words

which will form the vocabulary of the user and therefore will become key abstractions in your system and analysing to find these key abstractions is the process of discovery but once you come to the developer side who will implement the system, the developer will need to often deal with in addition to the key abstractions identified.

The developer will also need to deal with various different abstractions like database, like screen managers, like lists, queues and so on so forth because the implementation will happen in the domain of the developer which has a different set of vocabulary and identifying these key abstractions is a process of invention.

And naturally what the actual design process involved is how do you may these key abstractions into these different key abstractions that the developer will need to deal with, that is basically mapping the problems into the implementation and that is the role of the process of identification of key abstraction and as ever it is a highly domain dependent and quite a complex process and we will need to go through that more.

(Refer Slide Time: 22:46)



So to be able to apply these techniques and please be reminded, what I said at the beginning of this module this now we are trying to talk about techniques which more than theory will have to be practiced by us from the next module onwards, we will start applying on an exercise and try to see how things go. So what is most important is as you recapitulate on the approaches that we have talked of please realise that the, if you are given the problem of identification of objects and classes that the first thing to do.

Then first what you should try is to, the first you should try to go with the properties relevant to a particular problem domain which means that the first approach that we are taking, so you try to learn what are the properties that exist in that particular domain, what is the typical vocabulary and all that and try to do a clustering based on those properties.

So you are trying to follow a structural clustering and identifying the class structures and behaviours that exist in the system. With this you will be able to make some progress, some of the concepts abstractions will get identified but it is likely that this will not solve the whole of the problem. So it will fail at a certain point.

It will not give you a satisfactory class structure possibly and if it does not succeed, then you go to the clustering objects by concept. Then you try to identify that from amongst the vocabulary again that what are not primarily properties but more like concepts from the domain and let us try to apply those concepts and that will give rise to the behaviour of collaborating objects.

Because concepts often are not just like, I mean if they are very concrete and with respect to only one object or only type of class then, we will be more like properties but concepts are more across different objects, across different classes. So they will show up more as collaborating objects, behaviour of collaborating objects and that is what you should try to discover from the conceptual clustering which is your second step.

A few things are still unresolved if both these approaches have not been able to completely solve your problem, then you try to approach classification by association which is the basic approach of we are going by prototype, that you try to see okay. This is, I know this is a chess and I have a Chinese checkers and let me see how that associates with a chess, a soccer, a wrestling or a swimming and based on that association.

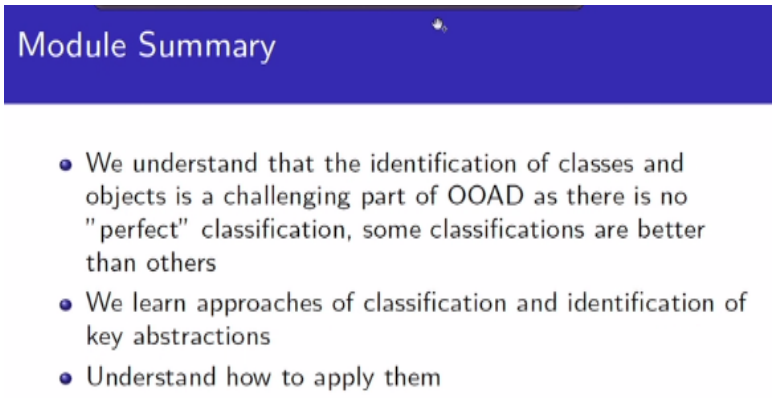
I will classify the concept and I will get a, I am using that prototypical object, I will get a classification of the concept and by this hopefully, you have been able to classify by this three. We have been able to complete your classification process and you have identified all the classes that you could do at this stage. So once, this is done, then you should engage in identifying the key abstraction that is you go deeper in the problem domain again.

You look into the vocabulary more closely and amongst all the different possible, you know candidate classes that you have got. You try to identify which are the key abstractions which are the most important one, the most engaging ones and try to pick them out and then look for more of their relationship to go forward. So this is the broad starting approach and again please recall that what I have been saying very repeatedly is that the whole process of object oriented analysis and design is a iterative refinement process.

So you start somewhere, apply some of the methods, make a hypothesis, make a design, based on that design you try to check how are you doing in terms of the quality of that design and then you try to refine the design. You try to use some more knowledge. So to extract some more knowledge, if required you go back to the customer and talk to the customer again.

Try to get more information possibly go the domain documentation, domain knowledge expert, get more information about the domain and again refine the design that you have done, again may be go over the whole of structural conceptual clustering and prototype been and re-identification or refinement of key abstractions discovering more of associations and then you come back and do the measure. So this repetitive process is very critical in terms of doing this design.

(Refer Slide Time: 27:56)



So in this module, we have understood and tried to understand the identification of classes and objects and we have observed that there is no right answer to this question. There is no absolutely proper way to identify and organize classes and in many cases one between

competing designs, one may be better in certain aspects and one may be better in certain other aspects. There may not be a clear winner.

But certainly there are various aspects, various measures and representations with which we will be able to say that this is a more proper design than another one. So based on this, our next task as we go into the module 17 onwards, we would now try to with this tools at our hand will take up the leave management system exercise and we will try to apply them and see actually on the ground how we can do this clustering, how we can identify the key abstraction, relationships, associations and slowly start building up the design.