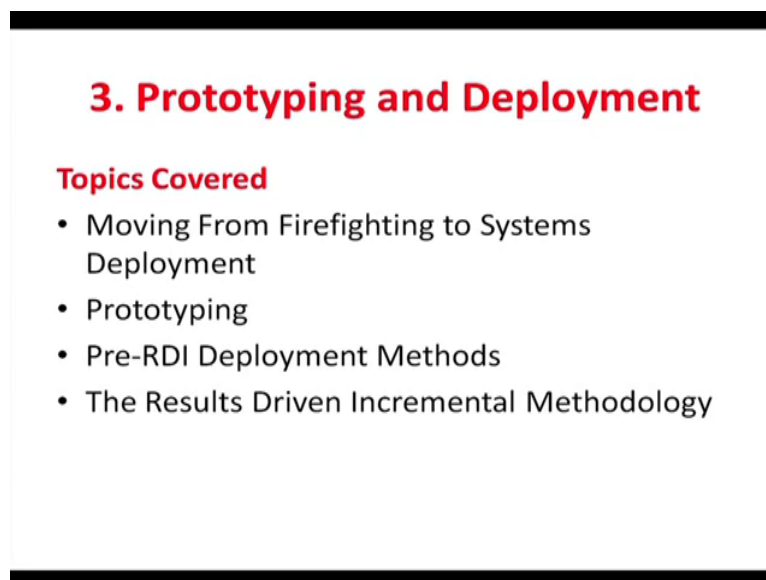


**Lecture – 24**  
**Prototyping and Deployment**

Okay so now we have discussed about the different layer of KM system starting with interface layer through depositories layer, so once you have decided about different layers and how they are going to be integrated then the next stage that is that we need to create a prototype and then finally we deploy the system, so what we are going to discuss here is that how to create a prototype of knowledge management system.

**(Refer Slide Time: 00:49)**



**3. Prototyping and Deployment**

**Topics Covered**

- Moving From Firefighting to Systems Deployment
- Prototyping
- Pre-RDI Deployment Methods
- The Results Driven Incremental Methodology

And then how we are going to deploy that particular system part of that we have already discussed, but here we are going to discuss it in much detail at how prototyping is done and then how we go about I move about system deployment so the topics that we are going to cover in the next two lectures include that what are the activities that we need to do for system development because since so now we have been talking about what are the things that need to be done in order to develop a good KM system.

Then once we have decided about all that all these things then we move to develop a prototype, prototype is nothing is but it is a kind of kind say replica or you can say man blueprint, which takes till tells you that okay this is this is the kind of KM system that we want to develop and then we will also discuss the deployment in, that how we go about

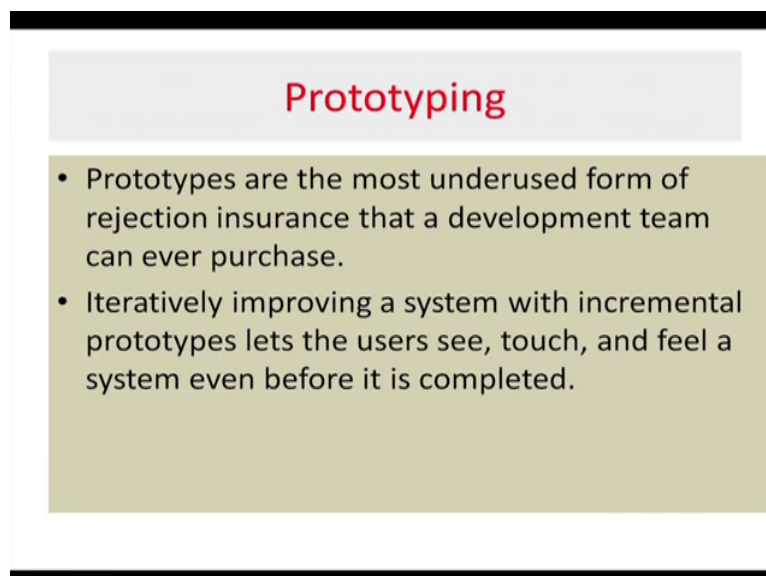
deploying a system whether it is going to be incremental or we are going to have a system, full fledged system.

But most cases it is recommended that you go for it incremental methodologies where you are going to develop one by one and add to the system and then ultimately we have a full-fledged system, instead of developing the full face system at one at one go but there is a need that you go for a incremental methodologies, we are going to develop one by one part of the system and also look into the result and that is why it is known as result driven incremental methodologies right.

So these are the topics that we are going to cover in the next two hours so we will start discussing about the prototyping and now if you look at system development see their lot of cost that is associated in the system development, because you have to see that it could be related to the technology costs, like having systems very true IT, software and hardware, a hard ways but apart from that you there are cost religious deployment and implementation of the cost system right sorry there KM system.

So it is not only other costs but there could be the cost could be divided into two part that is technical cost and non-technical cost in deployment and implementing of policy management system and you know when you are going to develop a deploy new system, it is always a learning issue because you learn from what you do so it gradually, you try to build up a system which is more robust and work more effectively to the advantage of the organization.

**(Refer Slide Time: 03:24)**



### Prototyping

- Prototypes are the most underused form of rejection insurance that a development team can ever purchase.
- Iteratively improving a system with incremental prototypes lets the users see, touch, and feel a system even before it is completed.

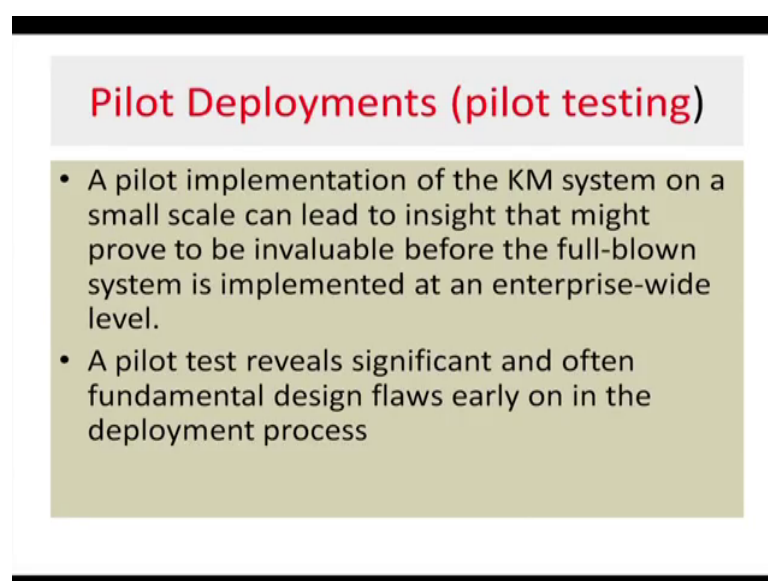
So we start discussing about what is prototyping? Now if you look at prototyping through types are basically what, if you look at prototyping it is nothing else but a kind of say raw document which tells you that okay, these are the things which are going to be included in a KM system. Now if you look at this it says it are the most underused form of rejection insurance, that a development team came over purchase, what does it mean?

It means that when you are going to buy insurance what happens, you are going to see that what are the benefits okay, how it is going to help you? Right what is the cost? What are the benefits? Right so similarly when you are going to buy or build a KM system we are going to give a analogy with the insurance that okay in what way it is going to benefit the organization and at the same time you also need to come is the management by showing them the prototype are the kind of system that you want to have okay.

So that the management is convinced about the benefits and use of the system for the people in the organization, so you need to develop a prototype, how are going to develop it? So you have a small part of our system KM implement in one part or one unit of the machine or relate to one particular function and then you keep on increasing that or extending it to other part are units of the system right depending upon your experience.

You can see whether it is working well whether it is effective, which is giving beneficial results to one part or the unit of the system okay and then gradually you move up I try to have a good system knowledge management system.

**(Refer Slide Time: 05:34)**



**Pilot Deployments (pilot testing)**

- A pilot implementation of the KM system on a small scale can lead to insight that might prove to be invaluable before the full-blown system is implemented at an enterprise-wide level.
- A pilot test reveals significant and often fundamental design flaws early on in the deployment process

So when I am talking about prototyping is nothing else basically you want to see whether the kind of system that you want to develop is going to really help you or is going to be beneficial advantage or not, so it developed one part okay and see that how does it help you, what are the results that you are going to achieve right. Say for example when you do a research what we do? We want to test the symptom on a small set of population right the idea is that if the assumptions are found correct on a small set of population, a small sample then you tries to move to a full-fledged research.

So that is what we know as pilot testing in research similarly when you are going for pilot deployment a pilot testing of a knowledge management system in an organization, you do not go for developing a full-fledged system in the beginning itself, the idea is that have a small system in place knowledge management system okay, try to implement in a particular unit look into the benefits okay and if you find that it is good it is providing this results then accordingly you try to develop a full fly system.

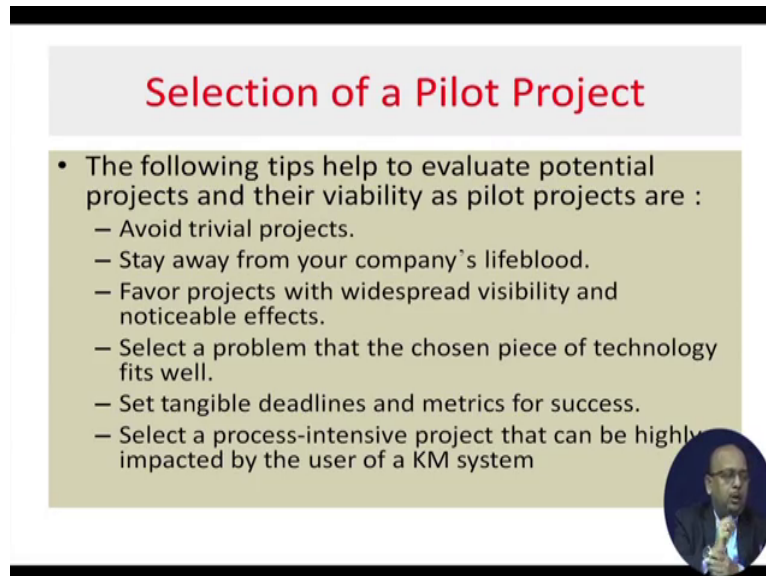
Let me again give an example like having an enterprise resource planning system, ERP system in an organization, say in our system in our organization we have now a full-fledged ERP system but before developing you full-fledged ERP system this system was tested on a small scale with a set of or a unit of student population, to see that whether it is cost-effective which is providing relevant information with that the people are able to use that system properly and effectively or not okay.

So once you are sure about this then only you extend it to other stakeholders like faculty, staff, and then you also integrate data related to various functions right, for example in organization you can start with you using the system with an R and D systems and see with the R and D systems which is basically more knowledge driven and knowledge intensive system whether it is really working there or not so if it is really working there because game system is basically used by R and D to come out with intuitive products.

And systems so if R and D any systems is finding it useful then you can extend it to other part of the organization so that they can make use of the KM system, so you can extend it to marketing, HR, finance and that and then you can integrate the entire system at the in organizational level right. The idea is that when you go for a pilot testing it is very important to see that whether your pilot test gives you significant and good results or not.

And that also helps you to identify whether they are certain attributes which are not working which are not functional or they are certain flaws in the design of the knowledge management system, before you move to a full phase deployment of a system okay. So there are two things you will be able to identify the flaws or the drawbacks of the system which can be pinpointed by their users okay and at the same time you can also look into the benefits of the system, before you move further right so you go for pilot deployment to start with.

**(Refer Slide Time: 09:13)**



**Selection of a Pilot Project**

- The following tips help to evaluate potential projects and their viability as pilot projects are :
  - Avoid trivial projects.
  - Stay away from your company's lifeblood.
  - Favor projects with widespread visibility and noticeable effects.
  - Select a problem that the chosen piece of technology fits well.
  - Set tangible deadlines and metrics for success.
  - Select a process-intensive project that can be highly impacted by the user of a KM system

A small circular video inset in the bottom right corner of the slide shows a man with glasses speaking.

Now how do you select a pilot project, because initially you have to start it from somewhere, so where you are going to start you may think about those projects are those and departments which are much more simpler more much say state which are not very, very critical for the organization, because if it fails it is going to hamper the organizational activities, so you have to use certain criteria when it comes to identifying and selecting a pilot project.

Because unless pilot projects are viable you cannot move up and extend it to other part of the system or the organization okay, so there are certain criteria like you need to avoid trivial projects which are very, very complex in nature because there you may not get the desired results right and there could be more disadvantages or drawbacks or flaws compared to the advantages that you can get.

Similarly you also need to stay away from companies line flood means that you need to leave out those departments which are very critical, because if it fails the entire activities is going to stand silent, then you have to look at those projects which are visible, which you know that if we are going to use these pilot projects in these departments are units or functions it is

going to really work. Say for example if you want to make use of KM system at the pilot level in the R and D system.

I think that is the best way to do it because if R and D system is going to make use of the KM system to go for innovations in the products and the processes that is going to be more principled and people will be able to recognize the effects they would be notice that okay I see the R and D department has a KM system and see that their productivity and efficiency has increase, they have come out with new products and systems a processes to improve the efficiency in the quality of our product of the organization.

So if these kind of things are coming up it means that people will come to know about it, so what will what will be the outcome yes people will look forward to these kind of things in their departments, so they may be pulled by these attractions of a successful pilot project in R and D system to other parts of the organizations, other functions of the organizations, similarly you also see that it fits with the technology.

Since R and D system is going to be a system where you are they are going to I am going to be more knowledge driven, knowledge intensive and they are going to use some and the kind of technology which basically favors a KM system, you can go for that and similarly you can also set certain deadlines in terms of outcomes or the times and then also see that how we are going to evaluate do you want need to develop certain metrics which I will discuss later for successfully evaluating a KM program right.

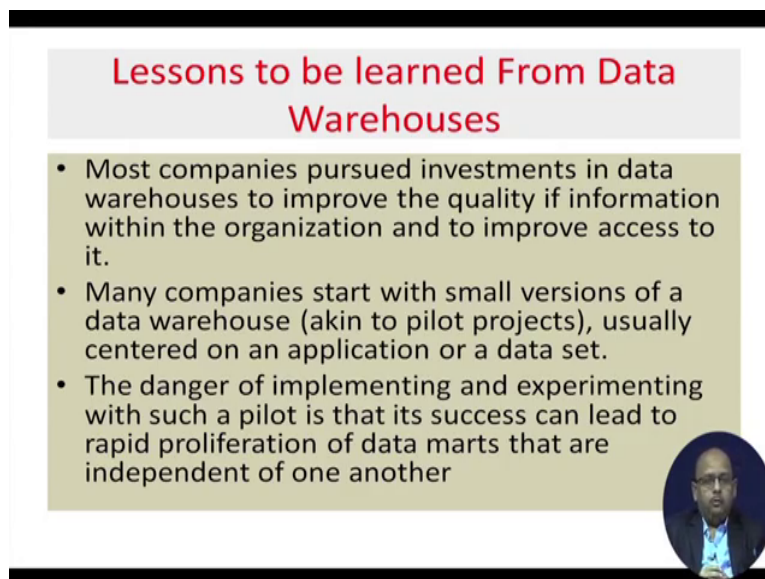
For example you can use traditional ROI images or you can also use balanced scorecard another kind of measures to see whether the knowledge management has really healthy organization or not and similarly you have to basically look for those projects which are not product oriented, but process for entity so when you go for process innovation R and D is the best example to look at it and that is where it the users are going to be impacted by the KM system.

Because if you are going to go for certain innovations or redesigning the process then they will look into the process and see that they can get lot of information about the KM systems about the different kind of processes which they can make use of it okay, so these are some of the considerations when you want to make use of a select a pilot project, so the idea is to that

you have to start it you find that okay, it is going to successful okay, it is not that critical for the organization because if it does not work it is not going to anyway hamper the activities of the organization okay.


And it fits with so far technology is concerned okay and you are able to manage and evaluate the success of the KM project, so you can see that okay in a given timeframe how will the R and D team has done up then in terms of coming out with the new products and systems are processes and that is how are able to manage the project work well.

**(Refer Slide Time: 13:49)**



**Lessons to be learned From Data Warehouses**

- Most companies pursued investments in data warehouses to improve the quality of information within the organization and to improve access to it.
- Many companies start with small versions of a data warehouse (akin to pilot projects), usually centered on an application or a data set.
- The danger of implementing and experimenting with such a pilot is that its success can lead to rapid proliferation of data marts that are independent of one another



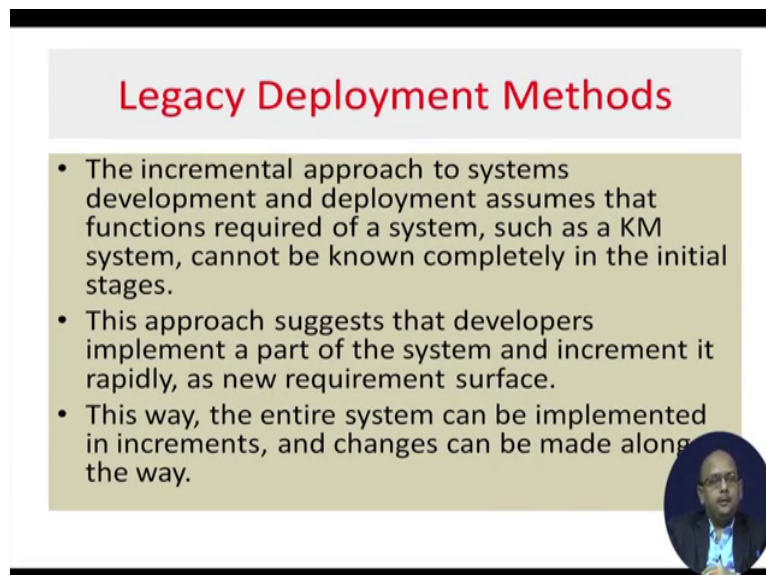
Now so far we have been talking about system deployment then you also need to see that what kind of data warehouses, we have okay so data warehouse is basically which are part of the repositories in the organization, so we have to see that instead of spending more money on creating more data, I use databases in data warehouses we need to basically focus more on the quality of information or the data that we have.

And then how we are going to have access to this so instead of basically emphasizing on the quantity of the data it is always good I prefer to have focus on quality of information and data that we are going to have in our repositories and then you also need to see that at the user interface level how learn to have access to this kind of data then at the next stage. You also need to see that for a small pilot project suppose you want to have a data warehouse and then especially.

For say for example in R and D case then the quality of information is more important than the quantity of information, that is relevant and actionable in the context, then since you are going to experiment with say pilot project having a KM system for a particular unit, so if you success if you experience success, then you can proliferate it to other parts and if you experience failure then that creates a lot of impediments are barriers for you to implement it to other part of the organization.


So you also need to see that when you are going to implement and experiment with such pilot projects okay ultimate goal is to achieve success and if you do not achieve success it will have its own consequences or dangers okay.

**(Refer Slide Time: 15:47)**



**Legacy Deployment Methods**

- The incremental approach to systems development and deployment assumes that functions required of a system, such as a KM system, cannot be known completely in the initial stages.
- This approach suggests that developers implement a part of the system and increment it rapidly, as new requirement surface.
- This way, the entire system can be implemented in increments, and changes can be made along the way.



Next legacy deployment method how we are going to integrate data which has the software other part that is already there in the organization with the KM system right see for example you have a R and D system which has gone for pilot project you have selected for the pilot project and then which is responsible for what product and process innovation and that is where you have deployed a KM system right.

Now how this KM system of R and D department it needs to integrate it with the system already existing systems your production department now the technology that is already there with the KM system need to be integrated with the R and D system, because whatever is being done by R and D system is to be used by the production system.



Because ultimately they are going to produce the goods and services, they are going to see that how the process innovation that is being developed by the R and D system has to be incorporated in the technology. That is there with the production operations department into some inefficiency quality features innovations whatever it is so it is very, very important that how we are going to integrate other systems with that or how I want to extend this to other part of the system okay.

And that is where you are going to use incremental approach this is basically for development and deployment of a KM function from one unit of one department a one or a department to and the department or in the organization, so what you need to look at is that yes you have to be very careful when it comes to legacy deployment because you have to see that in terms of technology and the things you are evil you are able to integrate them well.

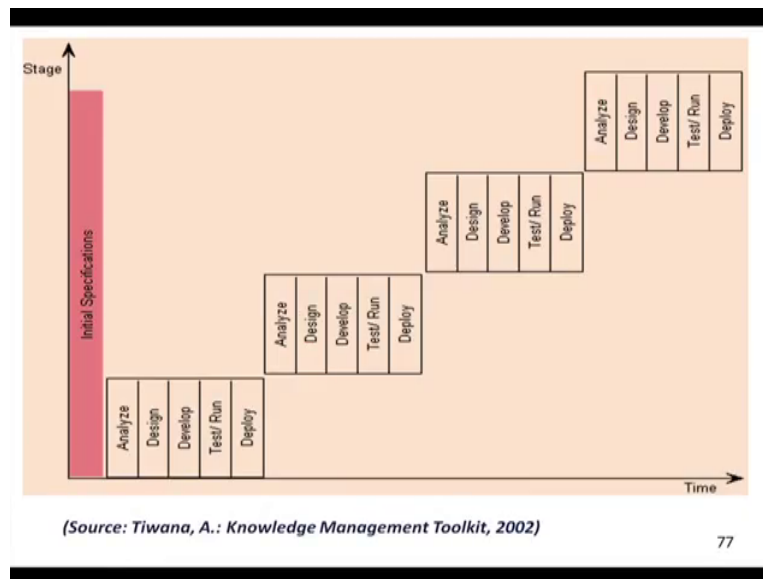
And if you are able to integrate them you will be able to implement the system in other parts of the organization you will be able to increment it rapidly depending upon the requirements, so for example if you gone to the in department, so production department had integrated the integrated the system with the KM system and that now it is the part of the KM system so whatever information within that goes to them.

And then it is embedded into the system in such a way so that production people are able to make use of the innovations that is made by the production R and D department based on the innovations what next, this then you need to integrate other parts of the system say marketing functional okay, because marketing people are going to sell the product, so they need to be integrate that okay this is the new feature of the product which is going to be advertised in the market.

So then marketing department has to integrate this with the production system, so now you have our R and D, production and marketing, these three departments are integrated next you feel that okay you have need to see that people have the capability to do these things, so you further move up to integrate the HR department, then finance department, is integral and then gradually you are going to integrate the entire organization together right.

And that is why the entire system can be implemented incremental and the changes made depending upon the requirement in the system right.

(Refer Slide Time: 19:03)



Look at this the different stages you analyze design, develop, test, deploy a system in a particular unit okay if you find that these these specification which is successful in a given time frame then move up to the next, then you move to the next, then you move to the next and you go through the same process and then you integrate this one by one, so that the system works well right.

(Refer Slide Time: 19:36)

### Legacy Deployment Methods

- The waterfall model, the parent of the incremental model for systems development, was the mainstay of the system development methodologies for years but has recently fallen out of favor.
- The waterfall model is a bad approach to take for implementing complex systems.
- If the feedback and learning loop are incorporated into this model and the project is broken down into discrete phases that build on another, it gives us the incremental approach model shown in Figure 12-2.

78

Then moving to the next part of legacy deployment basically the model that is used is known as waterfall, waterfall model you know what ultimately what happens the water is coming from somewhere it goes down and from where it moves to some other part right is not it, it means that the water that is there is collected from different sources okay, it gets a force and

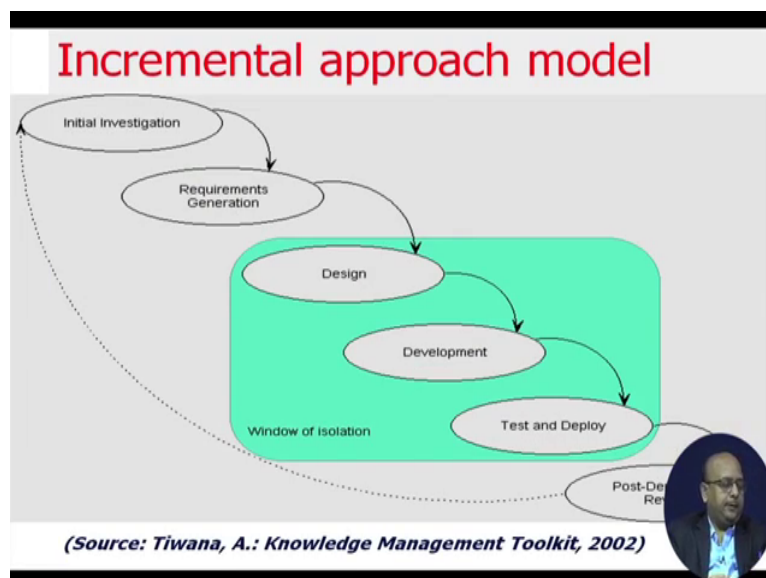
then in as a force it goes down right and from there again it moves to some other part in a different mode right.

So if you look at this waterfall model which is it is basically incremental because the water is coming from different sources gets collected at particular place, if a force is created then it moves further right and that is how you are going to use it this legacy deployment, so incremental by one you are going to deploy the system, one after another depending upon the success right.

But whether it is a good approach or not that is criticized because not necessarily to develop a complex system which is going to help you, because then the problem of integration and extension integration comes there, so what you need to know you need to create systems of feedback and learning at each stages okay, so this feedback and learning loop is very, very important because if you create that then what happens you create discrete phases and that each phases you learn from the experiences get the feedback.

And accordingly when you move to the next stage you make certain revisions in the system deployment so that the next part is better and correct right.

**(Refer Slide Time: 21:23)**



Now if you look at this, this is what happens this is the incremental approach that is to adopt it initial investigation okay you find what is the requirement then that is where you are going for design development and testing right, now if you look at these arrows you move from

here to this so this is a incremental approach in a pilot testing and then once something you have designed the system you have developed it.

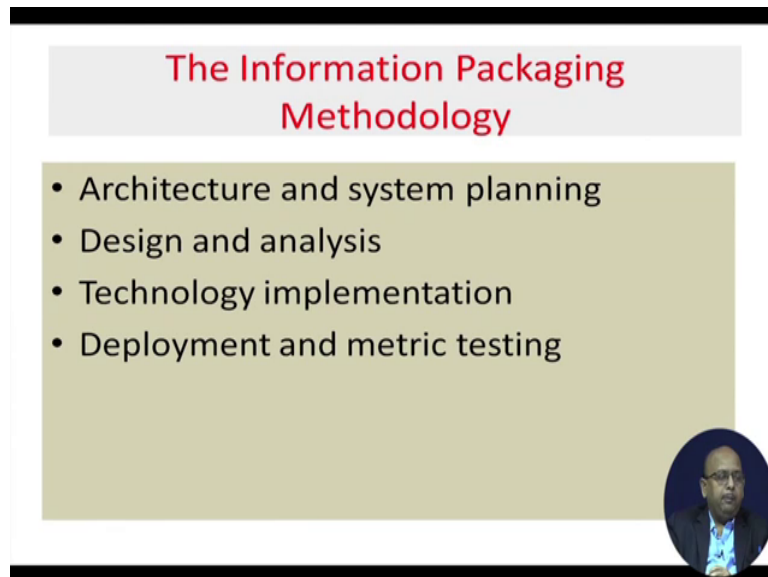
And then you are going for testing and deployment okay then you also go for post deployment at each pilot project, so once this suppose deployment true happens and you get the feedback you send it back to the, so that if there is any changes that can be made here and then when you go for testing in deploying it to other part you make the correction, the correction of the design development and when it comes to deployment of that in the next.

Say for example you have tried to integrate the R and D, within the production department, so the lessons that you have learned from R and D department can be integrated, we can basically help you to design a KM system which is going to be integrated the R and D in a better way similarly the lessons that you learn are the feedback, that you get from the production department and our R and D department can help you to design a better marketing approach right.

So when we are going to integrate our extended to the marketing section, you need to see that the learning experiences from the production department, and R and D department is embedded or is a part of the KM system when it comes to the marketing system okay, so that this feed feedback loop gives you information about the drawbacks and also the level at which an extent to which you are experiencing success, so that according way you can make changes in the design and the development aspects before I want to deploy to the next stage.


And that is where you are going to adopt a information incremental approach model around now.

**(Refer Slide Time: 23:33)**



## The Information Packaging Methodology

- Architecture and system planning
- Design and analysis
- Technology implementation
- Deployment and metric testing

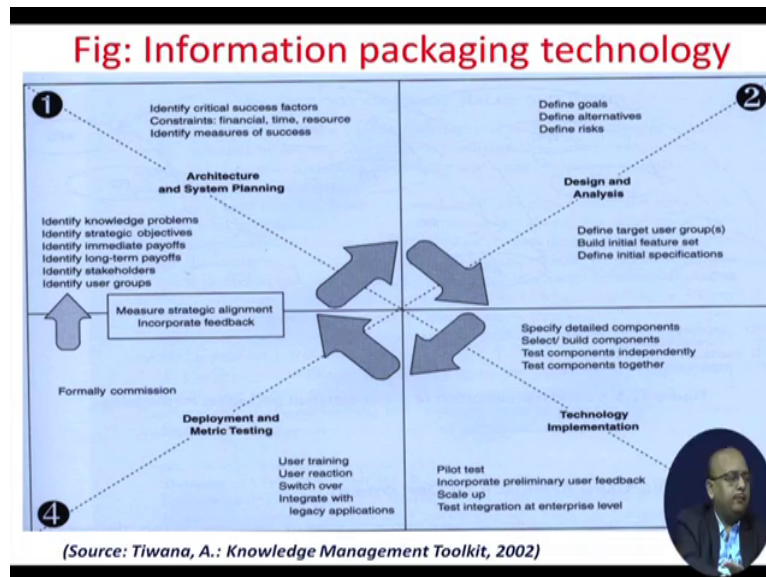


And the important issue is related to the packaging how we are going to package the information, so we have talked about certain things in the beginning also like architectural and system planning, what kind of knowledge management architecture have been to have? What kind of KM system you are going to have? So I already discussed about it like the KM system which will have different layers and how these layers are going to be integrated.

And what kind of knowledge management system or architecture is going to be here, we have also talked about the design and analysis part, how do you design the system, how to analyze the requirement at each and every stage and then how we are going to implement the technology, because technology is nothing else but these are the conduits of the information, so what kind of information and for technology in terms of software and hardware could be required.

And then how we are going to deploy the system and then test it metric testing means evaluation of the system right, so the information that is packaged into the content in the system mate comes testing that we need to follow these kind of things.

**(Refer Slide Time: 24:45)**



So that is what we are in to discuss one by one, see what happens at different stages the first part is the architecture and systems planning initial stages that is what we are going to do next stage, then for design and analysis then we are going to look into the technology part and finally we are going to deploy and then see that how the system is working and this is staged if you look at architecture in a system planning what actually happens.

You identify what are the success factors okay, what are the constraints in terms of Finance time and these kind of things, when the resources are available, how we are going to measure the success of a KM system then you are going to identify what kind of requirements are there in terms of knowledge, why how it is related with our strategy, what are the cost benefits in terms of payoffs and long term and short term okay.

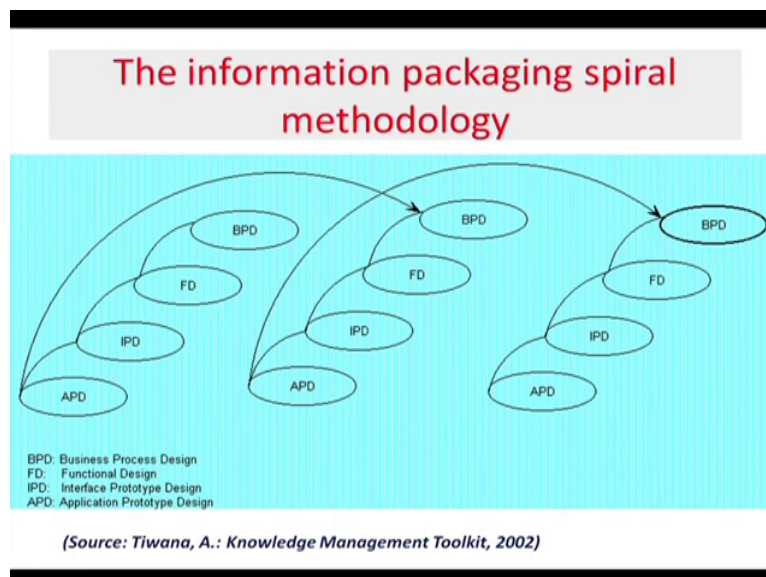
Who are the stakeholders will be using this knowledge right and then from there you move to the next part then you are going to define the knowledge, goal of the knowledge management system but and then you also look at the risks that is available there, then who is going to use it, what are the features and then you come out with initial specification at the KM system, moving to the next part.

That is where you are going to see that how the technology is going to enable that right, what do the different component of the system, how are going to select and build these components I am talking about the components of the KM system that is nothing else but there is seven layers of the component systems and how what is there in dependency and how they are going to be linked with each other okay.

And then finally you go for pilot testing and based on pilot testing you are going to get feedback and based on feedback, we are going to extend it scale up it to other parts of the organization and then you are going to integrate it at the organization level, now after the technology implementation part, the next is development and metric testing the plumbing and metric testing is nothing else but formally you are going to commission the KM system at the in travel organization level.

And then that is where you are going to provide training to the stakeholders, who are going to make use of the system you also get feedback from the various stakeholders and then we are going to integrate with other legacy applications, other software and hardware that is available with the organization, so that is how the information is packaged okay.

**(Refer Slide Time: 27:09)**

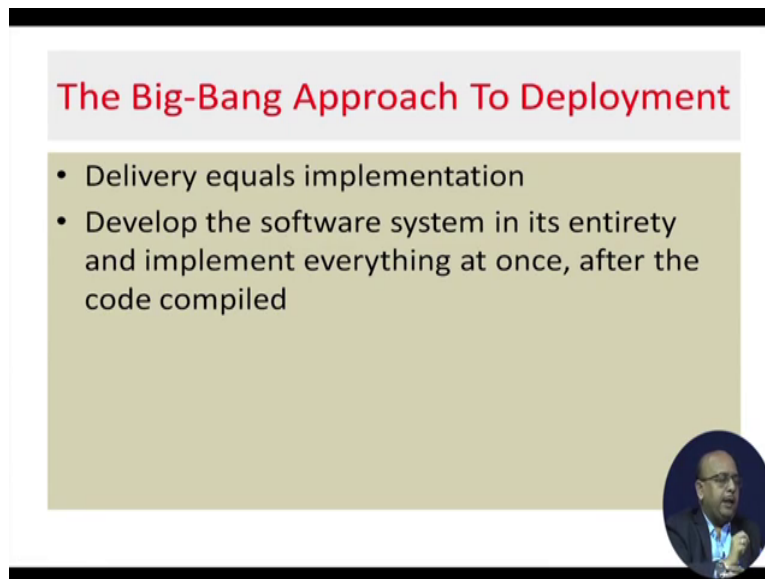


Now see what happens however into package this there is a spiral methodology that you are going to use one by one right, so starting with say business, process design and functional design, interface prototype, in the application prototype, see how they are related with each other, so it is a kind of a spiral that you have now build up starting with look at the business process, look at the function, what kind of prototype you want?

To have what is what its application and then again move to the next one, so that is how you move to the next one and then again move to the next and this is for different kind of system, so this may be far one, you need a department then you move to the next one, then you move to the next one and then you also keep on getting feedback, so that you can revise it

redeployed sorry modify the system at the next level. So that you have a better integration in the different departments right.

**(Refer Slide Time: 28:06)**



**The Big-Bang Approach To Deployment**

- Delivery equals implementation
- Develop the software system in its entirety and implement everything at once, after the code compiled

Then also have a big bang approach to deployment, where you are going for delivering equally implementation, means that you are going to do it okay, we are not going to follow say incremental methodology, but you are going to have everything for the entire organization, implement everything at once okay, once you have doubled in KM system we are going to make sure that it is going to implement it for all the departments at the same time.

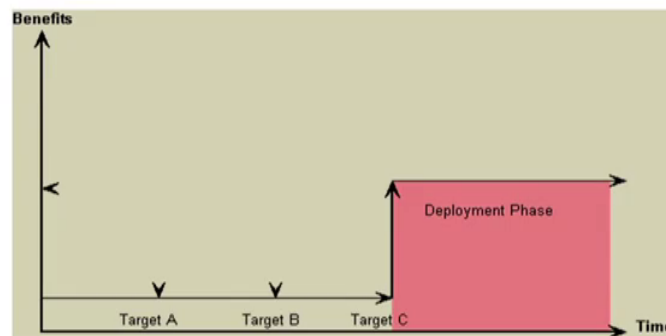
Now what do the advantage of having this kind of system, you can see the advantages are there like you can see that how the system is helping different groups are functions of people, who are going to be most affected, what are the drawbacks of different systems, you can get feedback from the entire system at the same at the same time and then you can go far go for making changes, but in incremental approach what happens you are going to make changes and integrate it one by one which could be there could be cost and time overrun.

Also but this is more cost beneficial if you look at cost-benefit analysis it is really bitter but the advantage is that it does not work then the entire KM system is going to fail and there is not going to provide you any kind of advantage, so there is a lot of risks that is involved in this kind of approach compared to an incremental approach, because that is less risky because you are going to use only one part of the system at one time right.

**(Refer Slide Time: 29:34)**



## The Big-Bang Approach To Deployment



(Source: Tiwana, A.: Knowledge Management Toolkit, 2002)

Look at this so what does this still look at the benefits and in the time period, so these are the different targets that you have and then you are going to deploy it and see what happens.

(Refer Slide Time: 29:56)

## Enterprise level Integration: Boon or Bane ?

- The flexibility is a boon because it offers intensively amplified benefits and abundance of functionality.
- But this boon is also the primary bane.
- The excessive flexibility mean that your have to tweak to work for your company, and this necessity changes a software introduction initiative into an organizational change initiative.

Then and the issue is that whether you should go for enterprise level integration or not, whether it is going to be advantageous for you or interesting to be avail right, see it is always good to be flexible. Because it allows you certain benefits and if you are not able to do it then you have certain risk that is associated with which can be reduced okay, but at the same time this could be a bane also right.

Because if you are going for an enterprise level integration, it means that you need very lot of flexibility in your system, lot of tweaking to work for your companies again and then you also need to see that the kind of software that you are using is good enough to take care of the

load, so when you can find price level integration it means that you are going to integrate the KM system at the organization level.

Then you make sure that using certain criteria that whether it is going to be successful or not like the kind of risks that is associated with this kind of system, what is the likelihood of success or failure? How the critical departments or functions are going to be affected by it? whether there is going to be cost in time overrun, so these are some of the issues that need to be lived in to see that they have to extensively amplify the benefits.

And if that happens then you can go for it otherwise not otherwise what will happen which is going to be occurs for you and this system is going if system is not going to work and if you are do not have the support system before these kind of things then it could create lot of problem for you so this is the enterprise level integration, thank you.