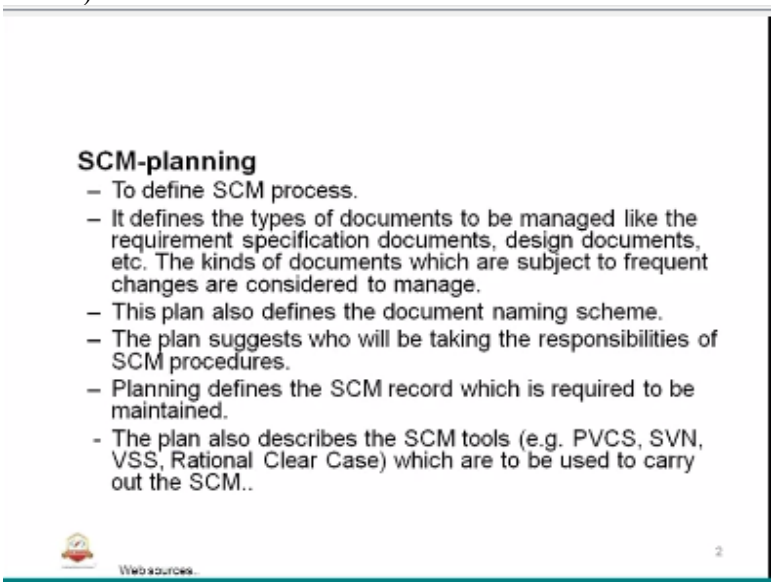
Embedded Software Testing
Unit5: Test Management
Lecture 3
Seer Akademi- NPTEL MOU

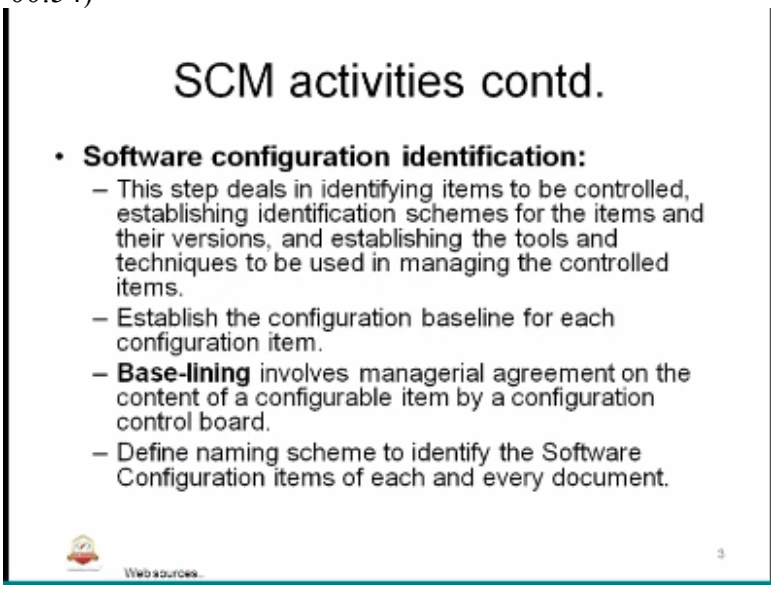
Welcome you to the next session of embedded software testing unit 5, is regarding the test management lecture 3 and today we studied about the chain management, and incident management SCM tools, a few example of contribution will have a look before that, we will try to see what we Have studied in the previous session, we have go through.

(Refer Slide Time: 00:39)



Some of the activities of the configuration management, in particular SCM software configuration management activities so it involves SCM planning, software configuration,

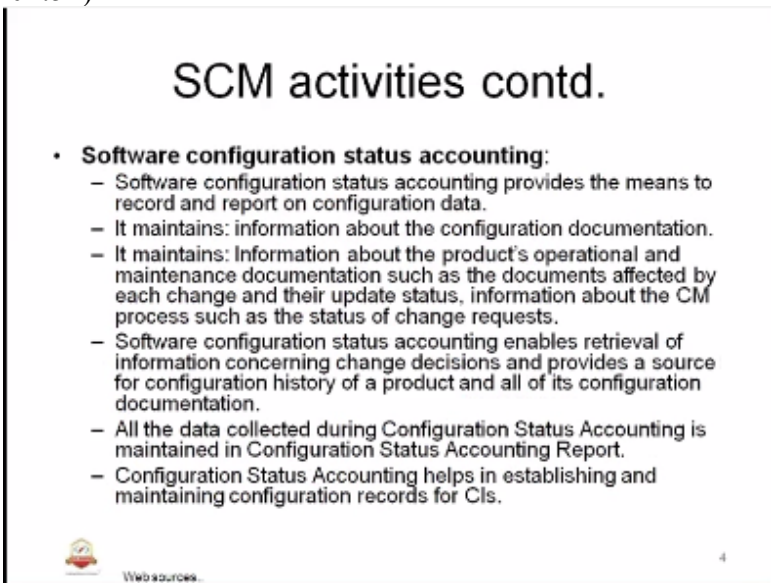
(Refer Slide Time: 00:54)



Identification, and term called base lining, and software configuration control, so basically, what we do here is, the life cycle changes, will manage a basically control, and dedicated the person

called this, who is called the admin of configuration control, basically, he can also be a member of who control configuration control board, it is called as, so basically they will decide changes and the impacts or what the kind of multiplication either to accept and reject or to refer it, all this will decide , so these are the important aspect of for SCM activities,

(Refer Slide Time: 01:52)

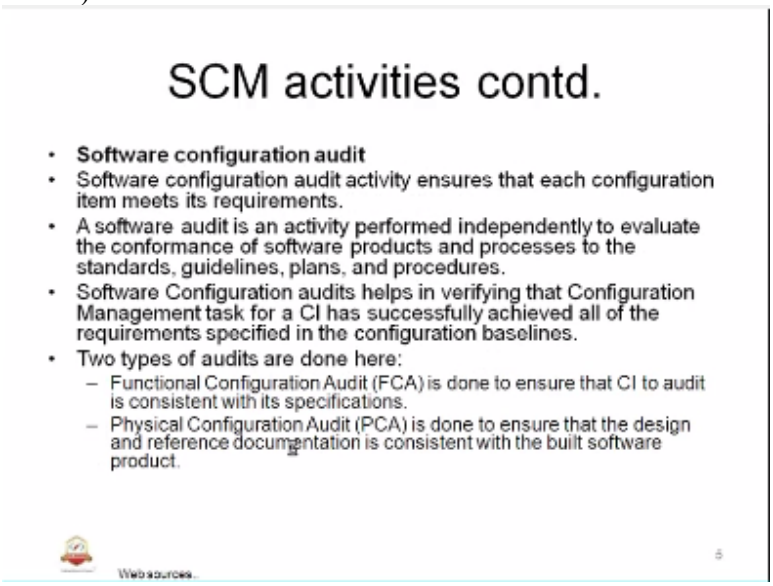


The next we will study about software configuration, status accounting, here basically see what we do his software configuration, status accounting, here basically what will do his, we will get the complete information about the configuration of the entire product, and check an item, or the configuration items, it should maintenance the information about the documentation, configuration documentation, basically you have a folders , where is a location, what is the base line, what is the region number, all the information about that there, in the part of the configuration documentation.

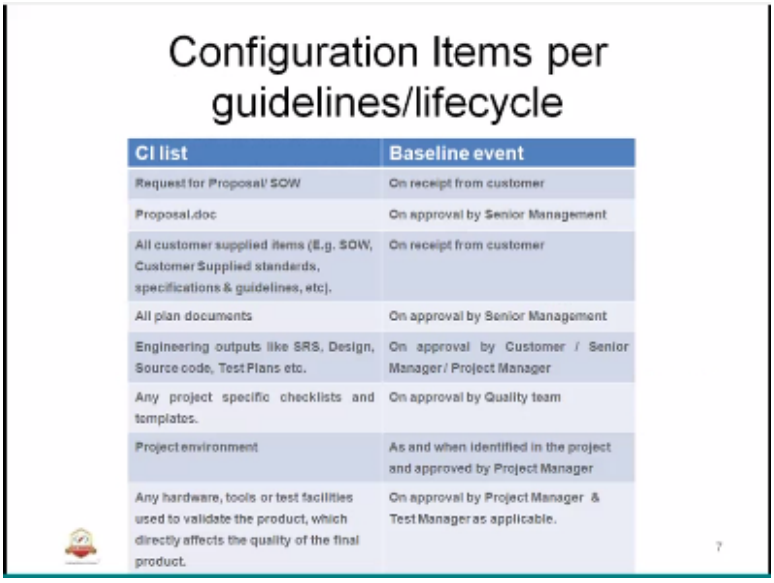
Configuration documentation should maintain the configuration status report, that status report is nothing but the actuators is called as software configuration status are contained, it also maintain the information about the product ‘s operational and maintenance documentation , how their inputs carried out, so what are the current information where developed, what are the documentation correcting by changes and those changes, whether they updated or in the changes have been progress, all this status about the configuration, each of the configuration identify the elements will be maintained.

And also it can able of retrieval and concerning change decisions and provides a source for configuration history, of a product, what is the history behind of all the changes and it is configuration detail will be provided by status of accounting, and already data collected during configuration status accounting is maintained in configuration status accounting report, so that is the frequency like, the report will be generated, this report generation also called up the configuration accounting report , basically it selects for them in a particular month, what where the configuration in terms are under grown changes updates, impacts all this information in the reported.

Configuration status accounting helps in establishing and maintaining the configuration records for class, the next activity.
(Refer Slide Time: 04:11)

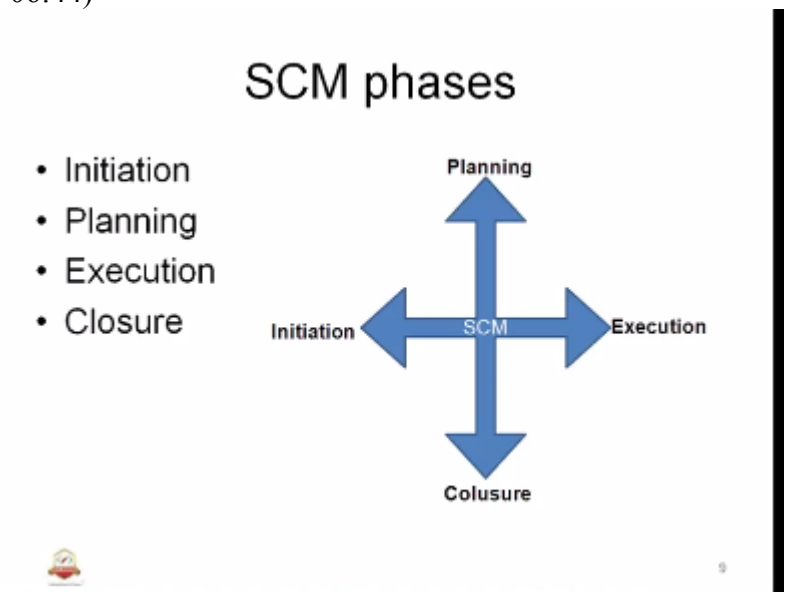


Is the software configuration audit, so how the configuration is accounting and CI is maintained so there is an independent audit, conduct by few auditors it can be same organization or independent or, outside or it could be a customer also, so this is an productivity, it processes the standards, guide lines, again at which they will go through and all these guide lines, standards, plans, and procedures will be a part of the two have, plan is nothing but what we studied about the SCM planning, for plan will tell, so what frequency, plan will be maintained, what are the CI is are all based, again to start whether these are clearly maintained and track the re assume activities are being audited, that is nothing but on software configuration audit.
So the software configuration audit performs independently the performance of software product process to be standard guide lines and standard procedures. so there are basically two types of audit which we have studied functional configuration and physical configuration PCA is done to ensure that the configuration item to audit is consistent with it is specifications, and physical configuration going to, make sure that the design and reference documentation is consistent with the built software product, so whatever we spoke about software and configuration and all those activities, physically available in the configuration , this part of the physical configuration hardware.
(Refer Slide Time: 05:57)



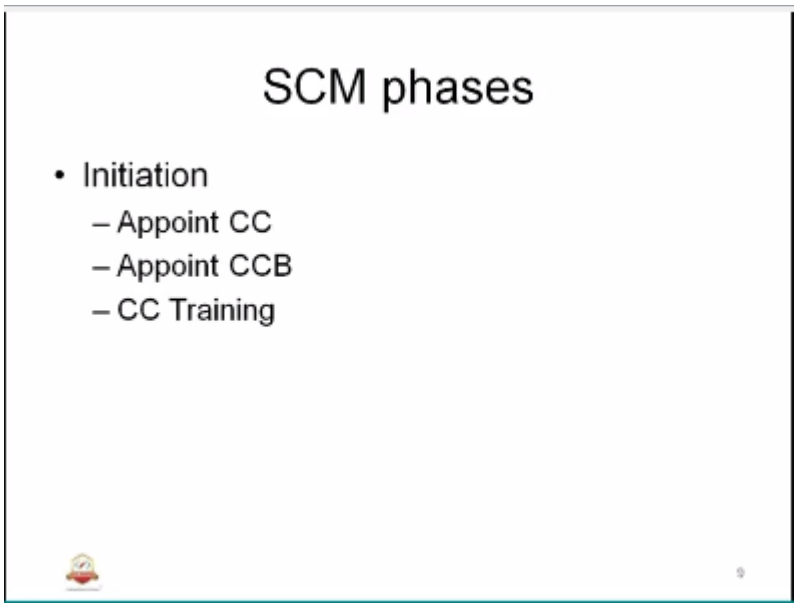
Then we will go through an example life cycle and guide lines various base line event are going to happen of that particular CI, for example a project line is called a CI suppose some new high tool give configuration data, some customer, some technical documents are arrived, that also in the part of the CI list and it has the own change, based on that, that is called the base line event and particular base line event will identify particular configuration item.

(Refer Slide Time: 06:44)



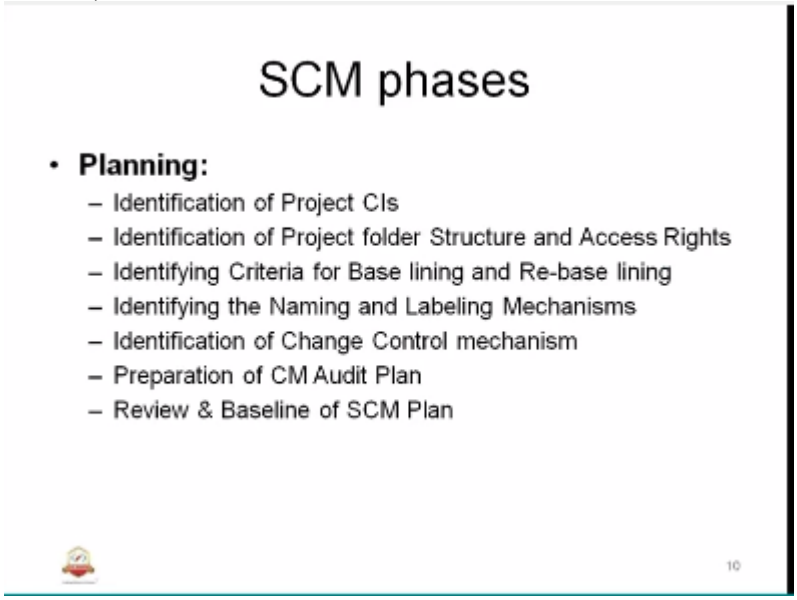
Then we went through the SCM phases, which are nothing but initiation, planning, execution, and closure.

(Refer Slide Time: 06:55)



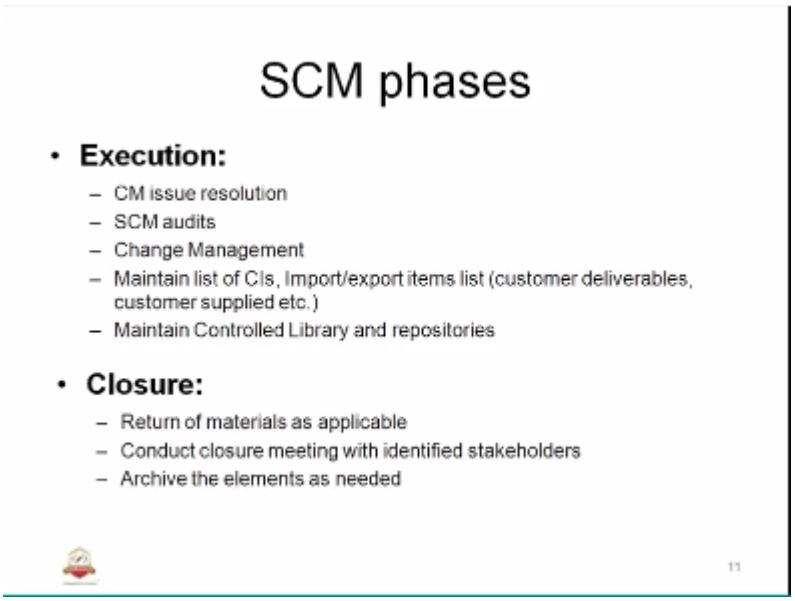
Initiation basically we start the project, we will do this, we appoint change control, a configuration control, about we also conducted CC training, as part of the contribution controller and CCB, CCB basically takes care of all project contribution detail, all is going to be maintained, it will be general as well as proceed to the particular project.

(Refer Slide Time: 07:24)



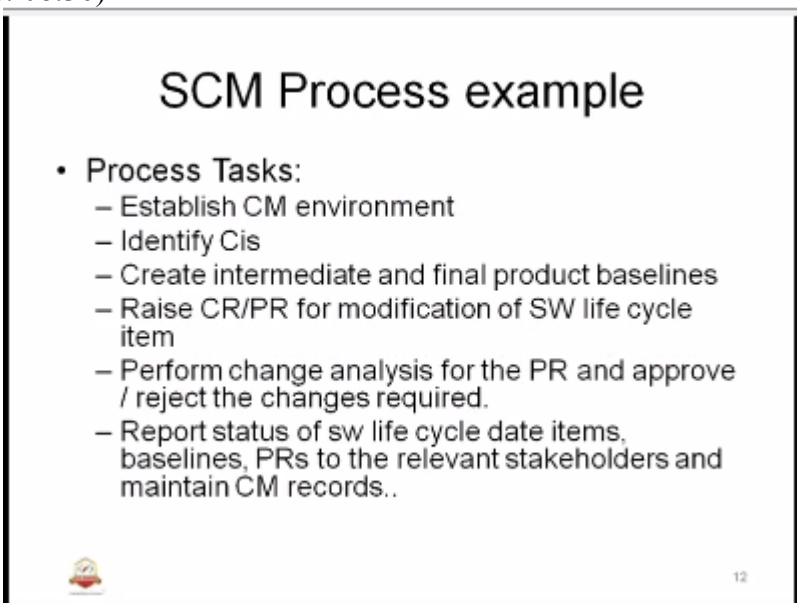
And the next one is the planning will be identification as CLS, we write the procedures and many communications , labeling mechanism, versioning will be on base line all this will be a part of planning, then will have the Execution.

(Refer Slide Time: 07:41)



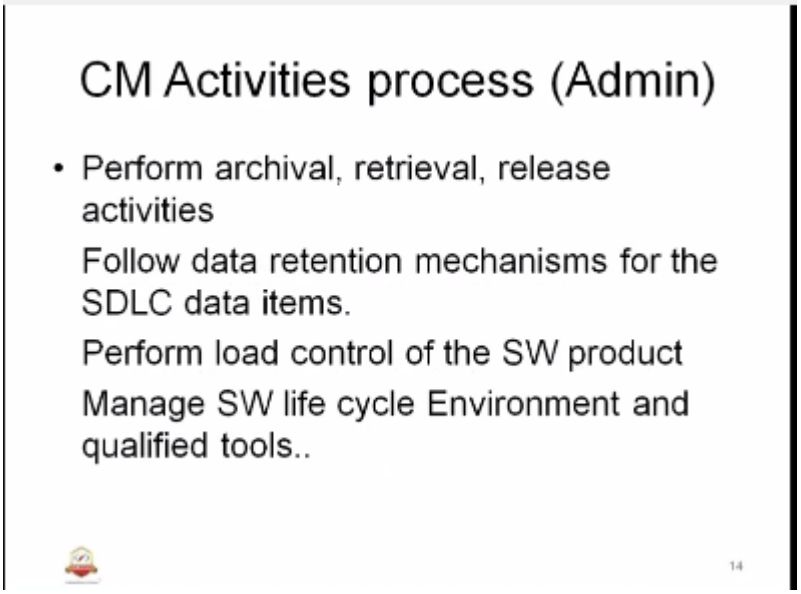
So whatever issues that have be go through during the CM, audits, we are going to correct it, correction and changes are part of the execution , closure is upon the same closure, basically it going to happen, when we have project getting identified and closure, and identify the closure in the , and immediate meeting, will identified the stakeholders, so what are the phases, that are going to, determine the identifiers and we are going to archive the data profiles or any repositories will relevance and any metals that we need to return back to some of SCM closure.

(Refer Slide Time: 08:36)

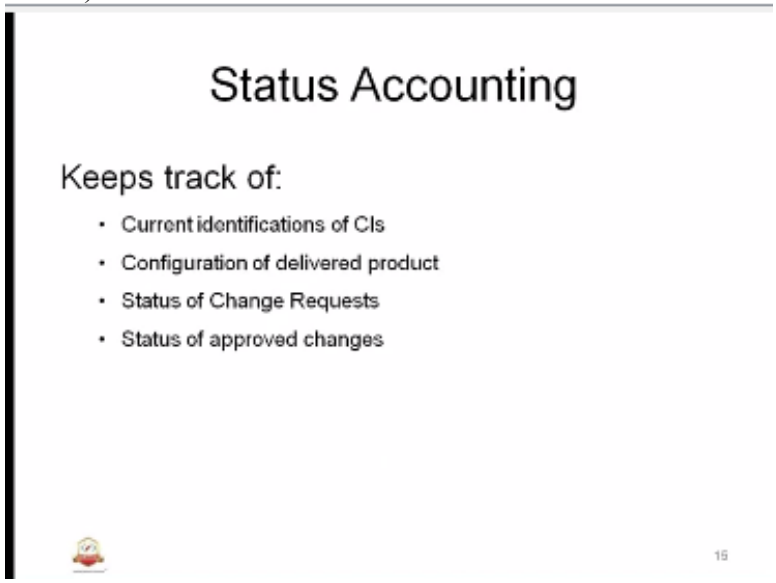


So also Example like objective scope what we are going to do for the same process, the process task establishing the CM environment and identify CIS, create intermediate and final product base lines and raise CR/PR for modification of SW life cycle item, CR is for changing request, PR is nothing but a problem required, changes could be announcement or new features PR is nothing but a problem report, due to an issue or error in the execution, and element of the software , this PR will be raised and PR should be approved or rejected by the relevant

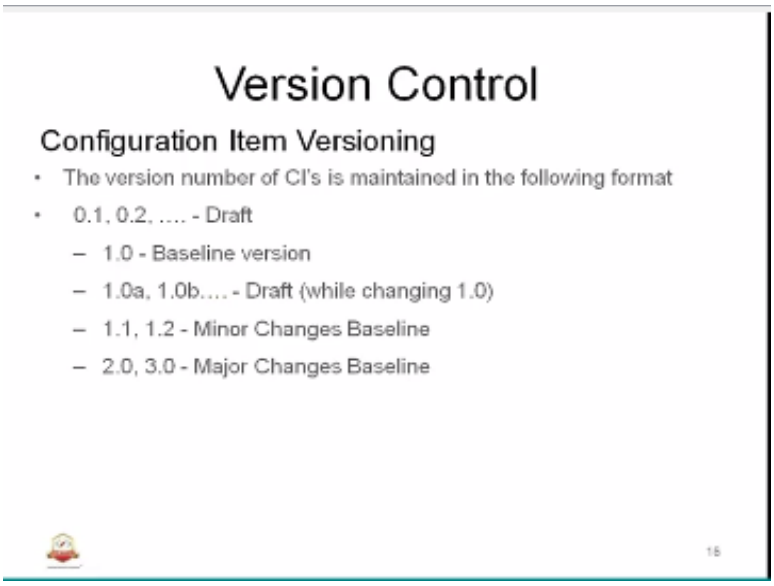
stakeholders and customize the changes will be analysis and imp actively approved by the segregate in the inventory configuration controller board.
(Refer Slide Time: 09:32)



And the dedicated person for doing the activities are projected is called CM or SCM activity who also like to called as configuration controller, who maintenance and do retention mechanism of SDLC data items, of SW maintain the life cycle environment and qualified tools.
(Refer Slide Time: 09:56)

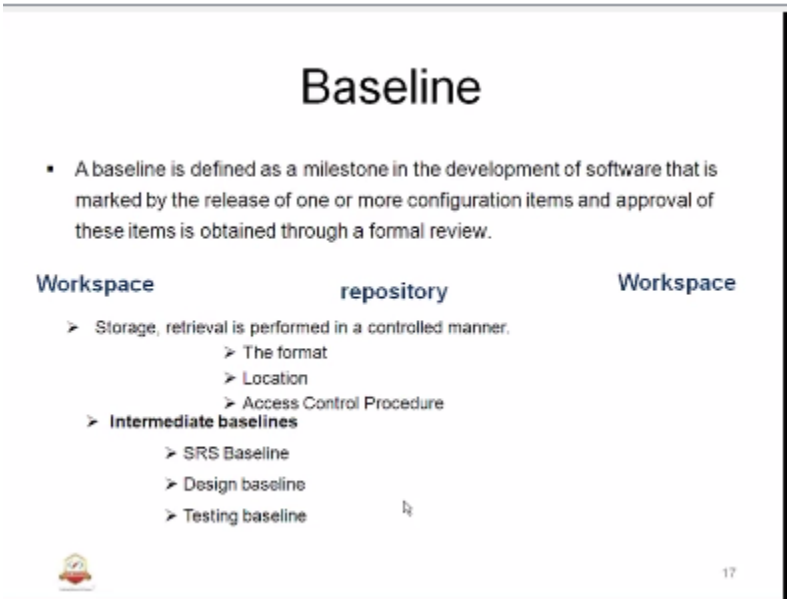


So status accounting keep track of CIS, delivered product, and we need to delivered, status of changes, what are the changes of request, and what are the approved changes.
(Refer Slide Time: 10:09)



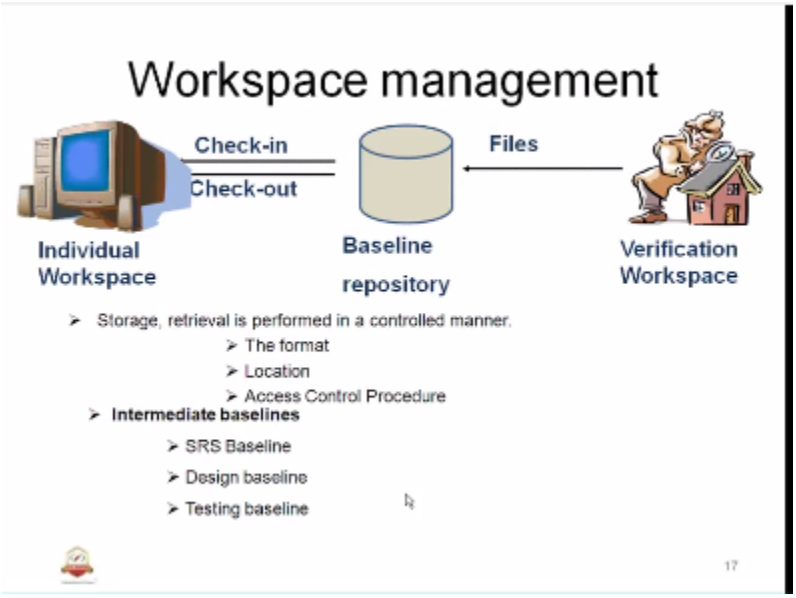
All this, and we end through, configuration item versioning, it has the different aspect, here we identify as the graph version, base line version, -1.0 base line, so graphs will be newly in terms of 1.0a, 1.0b, draft while changing 1.0), base line will be 1, 2, increments of integers, and any incremented versions are in to keep track of versioning 1.1.1.2 etc… minor changes the base line, 2.0 3.0- or 10. 0, 20.0 etc. it is depending on the project major are all under this version control.

(Refer Slide Time: 11:03)



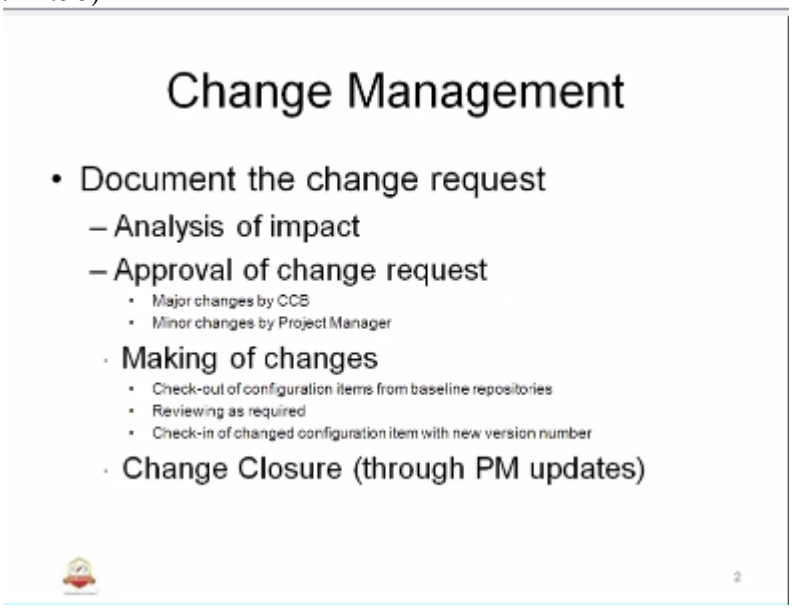
The thing that we have study about the base line minimum base line, and intermediate base line, will going to have it, this will be keep on happening during the project life cycle, so the project life cycle are SRS base line, design base line, testing base line, so all will be part of the configuration item.

(Refer slide Time: 11:25)



Also we are studied about workspace how going to be maintained this is the repository and repository, they will be used for adding the files, for taking out the files and updating, this processor is consequence about and the files are getting accepted or use by the verification, with the help of workspace how verification workspace, so that is what we said in previous class, we go through the change.

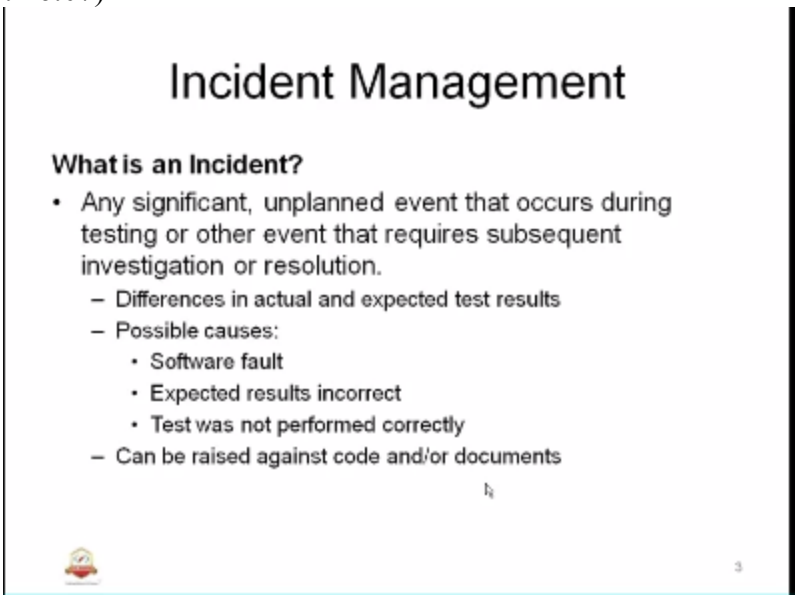
(Refer Slide Time: 11:58)



Management aspects in change management, so what is change, change means? any update or request for changes that are going to happen on a specific item that is something like living document, something like a living code, these all part of the change, on how we are going to manage it, or maintained it, these all part of the change management.

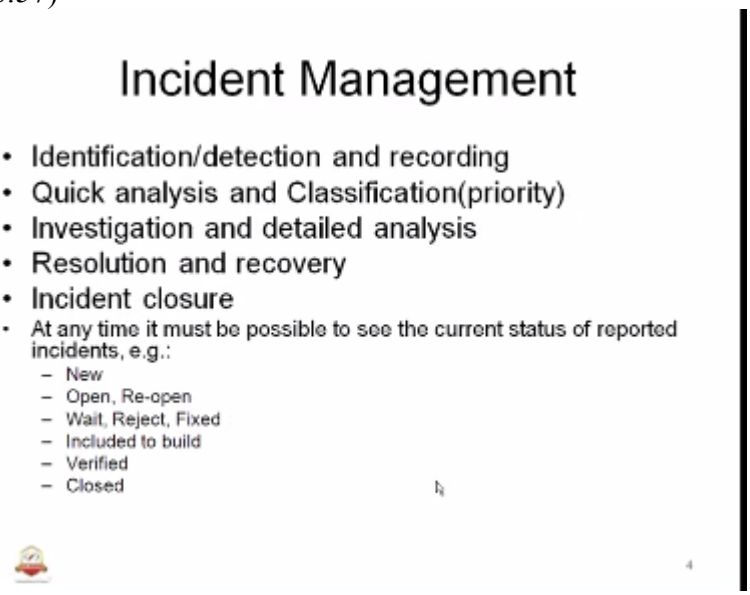
So we receive a request for change, that is called change request, first what we are going to his, we are going to documented and complete change, cycle change will happen, it is the help of below items, we analyze the interrupt of the particular change request, we are going to approve

for the analysis, whatever impact going to happen, all we are going to change, and that should be approval with the change request, and approval, those are nothing but project manager who is also can be the part of CCB,
And minor changes is of course the project manager level like can be done, day by day it has to go in the CCB here changes could be , a major functionality or a feature or any optimization etc..
Major changes are something like minor comments, fixing any documentation is resource called like an enough comments are not where etc.
So those kind of a, is called a category to changes, the minor changes, the major changes are called as category 1, likewise they are going to have different types of changes, accordingly we are going to get the approval by different team, one is by test manager, or project manager, with the help of the board, project manager change and board also can have the customer involved. So making up changes.
We do more element, element is already SVI we are going to check out or we are going to take it out the configuration in a repository, the repository maintained, CM tools like PVCS, SVN, some of the tools are also have MKS etc..we exactly, some of the repository tools that are used in the embedded automotive , and it have the relieve of changes required and those have done with the changes, and comments will have a re work, once a re work is done, it is good for change and we going to check with the new version number.
New version number we do not want to provide, the tool will take here, by implementing the next number, that implement number how is going to implement, they are going to define in the configuration of the CM code, while we are going to have a change closure, through project manager updates to the customer or it is a work book, or whatever it could be, so that is what it will follow in the change closure part of the change management.
(Refer Slide Time: 16:07)



Next so, what is an incident management, any significant unplanned event that occurs during the testing or other event that requires subsequent investigation or resolution, that means something has happen, let us define something, that something, this going to ask during the testing or development, basically we have focus on testing because of our presentation about testing, in

embedded software system, and that changes, that is a required, without that attend to an event , we cannot proceed.
There are certain lockage, because of that event, so that is nothing but in a incident, so that incident could be something like unit testing it can happen, or test case will be happen, environment could be happen or test planning could be happen or during the execution. We can get it this kind of test, likewise it is up to the planning, which are going to define the incident or if an incident get unplanned that also can be called as incident, under incident that is the take in care.
So something like actual and expected test results, that means is the same work, possible causes, software fault, expected result incorrect, test was not performed correctly, so the most change for this is that true incident? Basically because software test or embedded software testing is suppose to be done by the dedicated test and they have suppose to be identifying the parts of, develop code and accordingly requirement and whatever it is, when I suppose to take care of the, identifying faults and approach and test strategy cannot having the issues itself so it is difficult to manage of control.
That incident has to take here, so causes of incident could be a software issues or test the training or test is not done correctly or test was not performed correctly and incident could be raised in the code and or any documents or it could be environment issues, that is part of the embedded software testing so that is what called as an incident.
(Refer Slide Time: 18:57)



The next one is incident management so once we have a incident occurs, how we are going to manage, this what we have spoken in the management, so first we need to identifying, detect and record it, what is the incident, complete details of that, the occur of the event, whether it is occurring the frequency whatever surrounding conditions is happening? During what execution handling, what is the responsibility?
Which have been happened, likewise we have serve identification and detection changes of, that will be recorded, one start in the other line just going to capture in first step, then we are going to have quick analysis and classification, we are going to priority, suppose there are 3, 4 incident

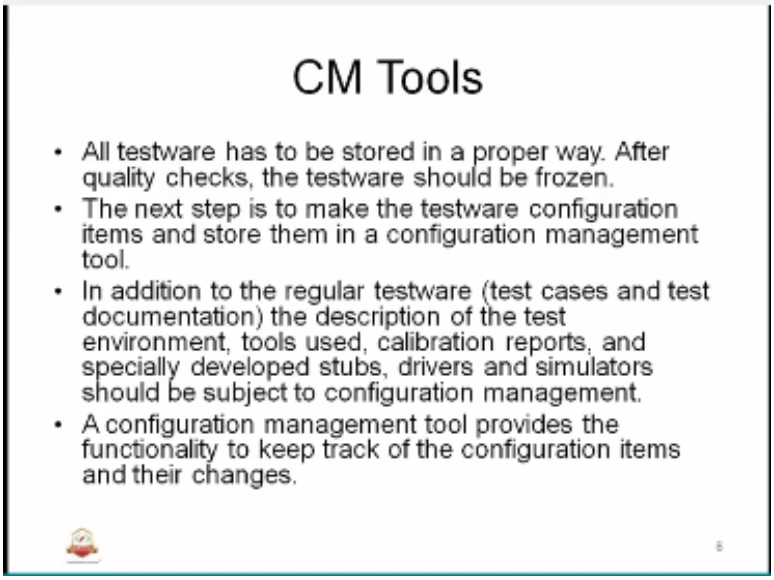
that have occurred designing on priority, will going to classify what sort of the incident it is? And it will come up quickly with quick analysis incident.
Then quick analysis will have an investigation, detailed analysis, so the second step is quick analysis, the next one is the detailed analysis okay, so once we have done with the investigation in a cause and the source of issues, it something like problem report with the possible solution also it would be a part of it, detailed analysis, actual be taking here, in the next step called the resolution recovery.
To recover the features is, that is what recovery in incident, and currently are called incident we have to address with the analysis that we have okay, and for this we have the resolution available and the incident no more, going to have a incident closure, and it any time it must be possible to the current status of the incident.
So once we have the incident is important it will go in to different stages, the stages are again they read from the various action that are going to be taken here, and it presents the incident, so incident will give 1 , we just open or re open the incident is done by the incident resolution, it is under the re stage or reject stage or reach the stage, the problem is fixed, so the next stage is include to build, one thing it be included , it has to be verified, where the incident is resolved, finally we are going to resolve it.
So all the stages can be seen in the mechanism of the life cycle of that, particular incident in that, so this is the part of the incident, so that we know that, what is the stage of the incident and this is about the status of an incident, we have the stages of incident where called a and reported, for example in terms of the incident is accepted with the stakeholders and incident is assigned to bolding, the incident could be on all hold.
Because there is no solution, temporary it is working or the issues will not be complained or the resolution has re term it is not consider to working so we can make it as a on hold sort of a data or this stage when the incident is solved and it is closed, so these are some of the use stages that also can be used for incident management.
So it hence what will happen in the resolution software system, suddenly some are issued or incident occur so definitely we are going to have a priority, so priority nothing but , here priority for the use of basic terms, the priority basically decided based on that, urgency of the impact, impact as well as the time line and it will require get re solved in a particular incident, so that is what we do the priority and according to all incident we will have a priority another thing explanation, matrix, execution t.
And incident which mind that incident, could be test issue or testing environment some were , hardware is broke up, it is need to the higher management and higher management could be solved to or talk to customer what are the input , so are will be part of the incident management, this matrix is called as, in such cases,
So there are relevant stake holders, they take care of the implement okay, any required that we are going to have, in terms of configuration in change management, or incident management or software configuration all this will have a heading called revision history, because this is the part of the configuration, so we need to identify or highlight the particular configuration , what was done during the various revision of the particular, here we have an example has to be there in minimally.

It any of occupies, it can be any design of matrix, requirements, test case are index, it could be split or it could be a software code, anything should have a revision, and the revision should have a minimum elements such as the below 4 or 5, or once it will have a revision number, it will increment, we are get in to revision, and the date , author will responsible for revision and who as approval for review, have put an example, and whatever changes that has under that, using that revision.

Like we have updates as spoken in to a new template, the new template, so that is going to have an another revision, that revision will be mentioned , likewise we are going to have a revision and one more thing, is non literal, it assume retaining on the , type of customer you have, European follow with the one , European, Asian countries have PD month and year so some countries have seen year, month , they have used : or /, or corner depending on kind of customer, so we need to take care of all uses.

It is very common, they are trying to shine with intendance that, may 5th they want to, may 28th they want to deliver something, suppose you have mention like 20, or they will maintain, so you could have mention something like a may 10th 2014 but customer thinks that the only, it is not touching the format, similarly you abort to have 05-10-2014 or our customer where they want to 10th may so these also very important, so we should take care of the this terms , we are interact with these customer in types of merely mentioning whatever revision that is taken here, in a particular configuration item, so that is spoke on accurately is very important , so the next one is the, we have afraid about.

(Refer Slide Time: 29:28)



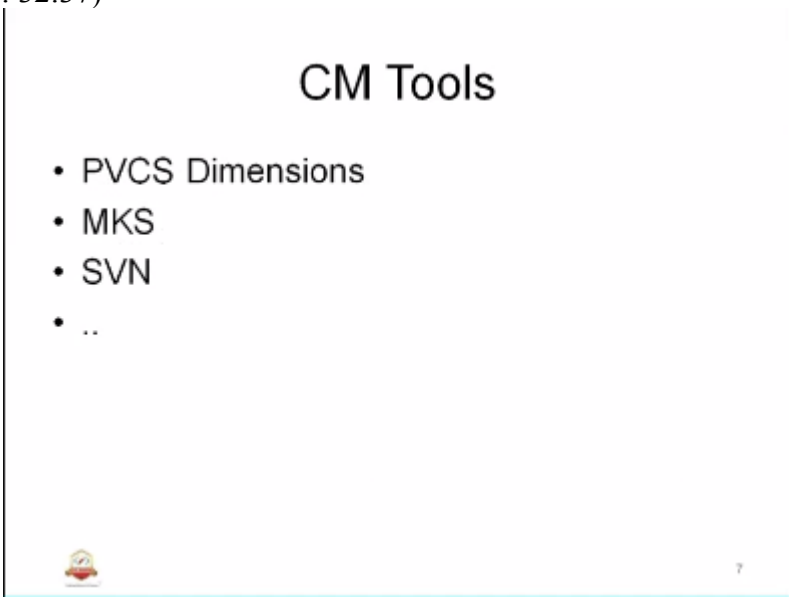
Incident management, software configuration management having the 4 phases, and 5 stages, five types of planning CI status, status and audit, all this have maintained, that repository is nothing but, a data is physical condition, how we are going to maintain the location, so we use tools, these tools are there which will take care of, this configuration management, that is what we study CM tools, all test ware has to be stored in a proper way, after quality checks, the test ware should be frozen. We know what the test ware is, all the attributes of testing, elements is the

part of test ware once the hardware is ready for some stage, it has to be frozen, move for the next stage.

The next step is to make the test ware configuration items and store them in a configuration management tool, that means using a tool to conquer at the particular location, or the server, in addition to the regular test aware, test case and test documentation, the description of test environment tools used, calibration report and specially developed stubs, and drivers and simulators should be subject to configuration in management, all this need tools in documents and genetic environment even the wiring or hardware changes , or hardware or you may definitely have a circuit, is nothing but a diagram or design, so that design is part of the configuration stubs, and anything picture regarding that PCP or any parts something like switches or the potential meters, these also can be on trigger with the specific version and ended all the details

Or if it is applied the customer, that should be configuration , all this will be part of the test ware and it need to have a configuration and management, with the help of configuration tool, a configuration management tool provides the functionality to keep track of the configuration items and their changes, so why should is important is? We need to select the track of each configuration, we are using it, and what are the changes that is going where, currently getting used in a code, which version of the CM tool, so all this has to be maintained, and that maintenance should be help of CM tool

(Refer Slide Time: 32:37)



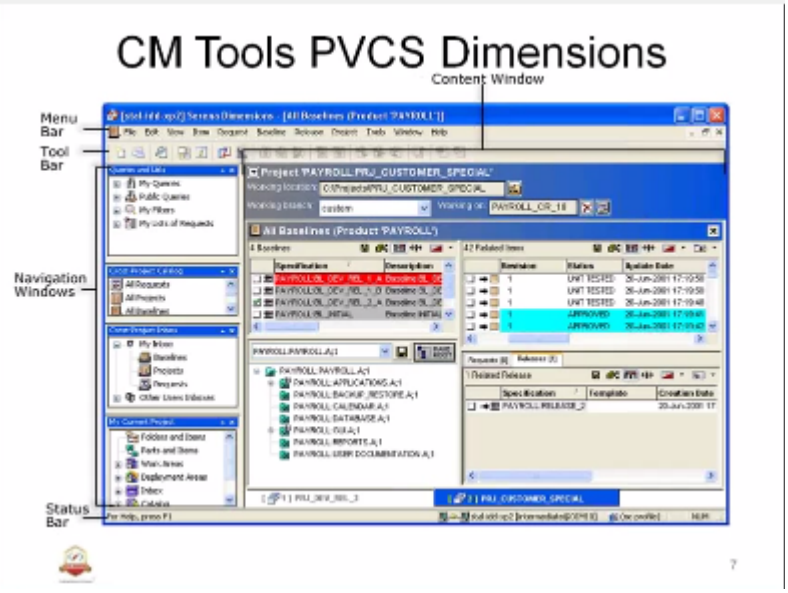
An example of CM tools, PVCS dimensions, MKS, SVN, etc… these are typically followed in embedded industry, such as uniform, above the, we have the various tools, derived case and all that, depending on each sort of customer and project that is being an, it also developed by the, documentation and some time it is may happens in such that, the configuration tool that the half surety will uses.

It could be different and what the customer has? So it will be strictly to maintained both the version of, to be customer, so we need to take care of such case will do the project planning for

the contribution planning, this will be co ordination with the customer, similarly it is difficult if you use the different version of configuration tools, that is also important, this cross the geography, that means we have server one place, client has different place, and work have different version of, a tool that is been used.

We need to take care of different versions, or we need to have the afraid on down grade of the production tool, for us to take care of differences, so that is the important aspect of the contribution management, okay, let us study about the little example of a CM tools, the tools that is called PVCS dimension, it is a web page, as well as is a standard application both are there, depending on person, they will decide version beginnings and it should be install, and is basically maintain all this configuration and versions history, checking and check out details adding the items.

(Refer Slide Time: 35:03)



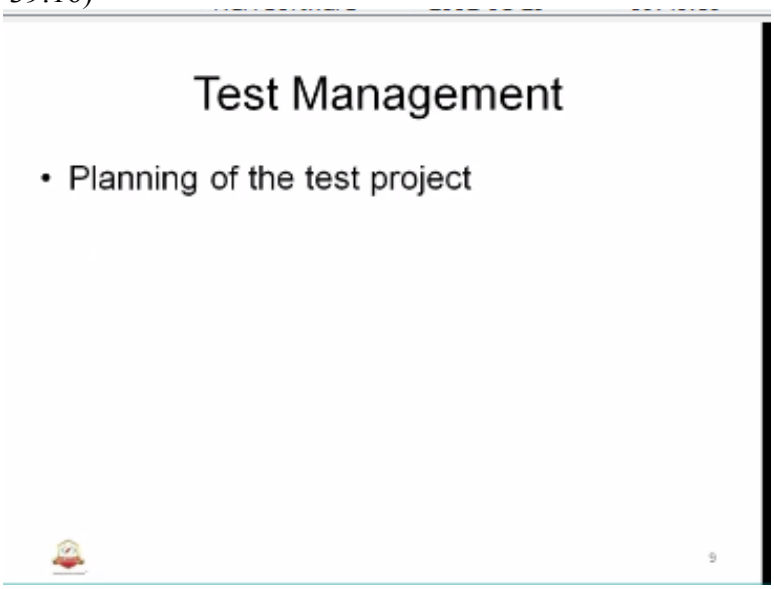
All this well be part of the active that active can take care with the help of the PVCS dimensions tool which are seen this are the menu of various options and this is an example of parole and this are tool bar icon and basically the left hand side of this tool we can see the folder all the folder structure where the CI configuration items are available and we may have in status of whatever the items we have used will viable in the status bar and on the right hand side we can feed the project name and project location working bench customer local whatever it is working on in which version working on is nothing but.

The a specification item which particular window is opened that is change request 18 or whatever it could be so any update to this elements or the configuration items that are going to be placed here definitely we will have some issue in terms of identifying it because of what or which I am going to use this tool which is are the invert it could be change request or a problem report or it could be a any failures that is resulting in using this tools for checking in or checking out on excess here you can see base lines and all the base lines are listed and each base lines can have sub baselines you can see par all applications par all.

Back up in store par all calendar, data base huge depositary way this can be maintained if you just local C drive or D drive to maintain all or you save it tomorrow I am going to make it for exile sheet and throw local modification it is very difficult in TDS wait 100s peoples 100 of software artifacts or 100s of proposal element if it is one or two defiantly we can maintain it and it is difficult it have the configuration items and another important that they need to retain it in terms of retain hey because the CIs suppose to maintained throw out the project as well as pos project in terms of warranty whatever it could be or a reuse for the other projects exporting, importing all this stuff.

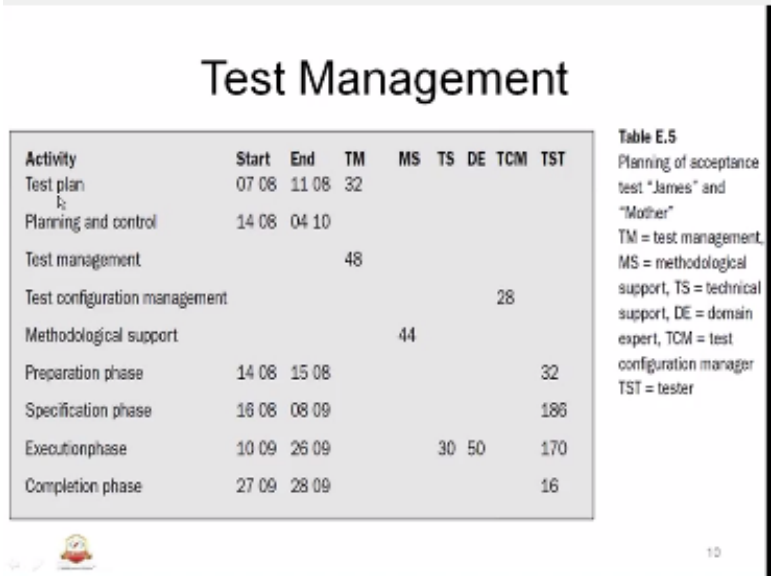
So definitely there is a need of configuration mange in embedded software the industry are identified by the 100 percentage manage the fellow it without which the client will not agree we need to have planning for this so this basically PVCS dimensions tool we need to have related project prospective items listed on the folders and each items will have intermediate versions or major versions or a baseline version all this have to be specified in the work instructions at according to the work instructions the configuration we are going to configured tools and it is going to take care throw the project okay, so that is about the configuration or the change management and incident management and the CM tolls.

(Refer Slide Time: 39:16)



Next we will come to the rest part called as test management so test management is nothing but, testing the project and how we are going to mange it so how you are going to mange it? Planning of the test project it is nothing but test management.

(Refer Slide time: 39:32)

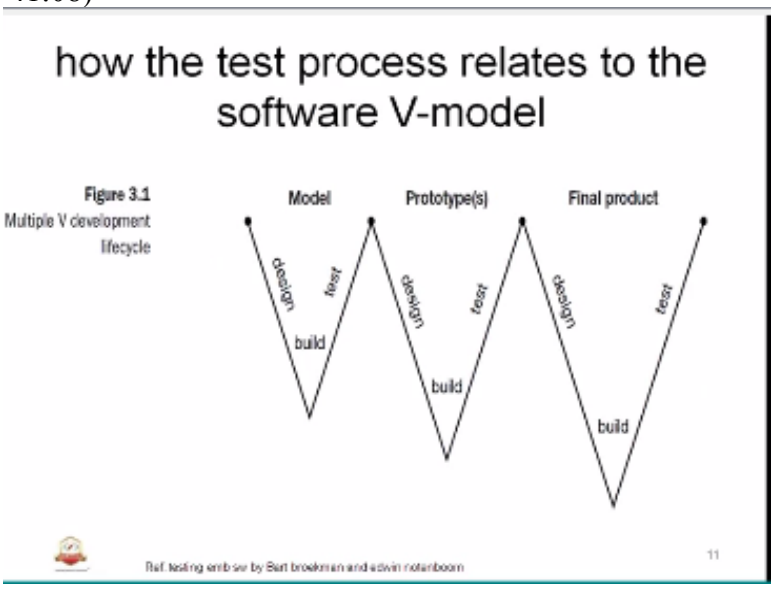


So we have test managements in terms of varies actives, active test plan, planning and control test managements, test configuration management, methodological support, preparation phase, specification phase, execution phase, completion phase so we are going to have a test management identifying all this artifacts in terms of planning of the ways of this items and we are going to identify.

The start date and end date total management effort that is test management CB abbreviations of each of this TM is the test management, MS is the methodological support TS is technical support this are basically the different test artifacts TCM is little bit of test configuration management TST is tester so which identifies the various efforts and that efforts during start on period.

So this are all the part of the test management basically this well be part of the test projects inactions and planning so we have done with this regular to manage the entire project with the help of artifacts even to track and maintain it okay.

(Refer Slide Time: 41:08)



So test process we know the test management we have and we have the test process which had it is won lifecycle model and we try to alien the test process in terms software remodel so in our privies management we studded about V-model nested V model all that so similarly test is also part of the V model and we take an example of this figure from that book from bar brakeman from this figure you can see the multiple V development lifecycle where we have modeled where we have defected three V multiple development lifecycle first one is the model second one is the prototype.

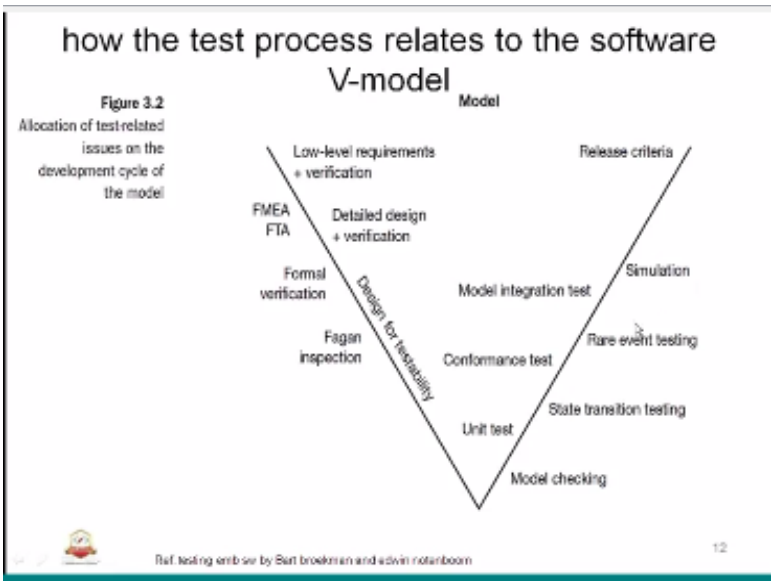
Third one is the final product in model we have designed build in interims of development in product we have designed and build and final product designed each of this well have a testing frames each of this well be in our testing frames So testing is also the part of this V so that is how the test process is closely related with the software V model this is important why because we are going to have the testing done along with the test approach or test scenarios how they are going to arrive.

A test approach are the test scenarios or there are with the help of the varies aspects of the development lifecycle those aspects could be requirements design and it is software build that is how the test process is very much close and related to the V model of the project lifecycle so basically the test process involves lot of test actives there are many design techniques that can be applied.

Test levels test types that must be executed and the test sheets required attentions the multiple V model ashiest in situation this activates dimension so by making this basically the test process multiple V model se provides inside A to the when one can start actives and one what is a time for best in terms of execution which test issues are most leveled active staging the development process.

So eelier we where testing in complex situation of multiple V model this is a bit complex task suppose my6 design is very tricky we are not sure about the build which is going to work the prototype sp according we are going to have the test process also little bit complicated so how we are going to mange this the test manger in overall pitcher of the relevant test artifacts or the active or the issues so definitely a bigger pitcher and testing management is very important in terms of getting into the various complex tasks of course it depends on the particular project is very unique to each project bur general principles of developing the test process always apply okay.

(Refer Slide Time: 45:50)

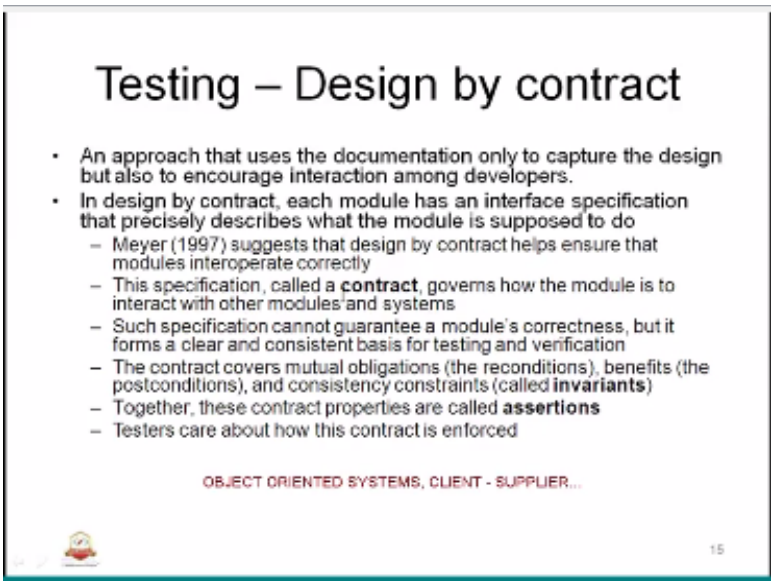


So next one it is the excision of what we have seen in the privies multiple V model V developments lifecycle this talking in detail interims of hoe we have a different formalistically V mode, model, prototype, final product has like model has low level requirements verifications and we have simulation and we have rare event testing state transition testing all this aliened with the varies lifecycle artifacts we can see in detail similarity for prototype we have low level requirements we have a release criteria we have a detailed design verification we have system integration test this all we have seen right, this multiple remodels.

So it is strange actually in terms of test process how it is related to software V model so we have a code review so for build the code review we need to have code available and unformatted we are going to have testing also for the compound testing the compounds testing requires submitted vermin codes with that we are going to have the code coverage analysis process part of the test process which is aliened to be V mode that is what it talks around okay.

The prototype also protestation or production formation is there verification requirements is there accordingly we are going to have the product certification mechanism , certification testing or excepted testing system expected testing, degrading testing for multiple version of the software are change in the software only more requirements or the design verification which is interact in the systems so that is how is allocated such way that the test process are aligned with the V model okay the next chapter is about testing design by contract.

(Refer Slide Time: 48:30)



So what is design by contract see anther aspects of testing which is part of the test management so the test management has to define the kind of testing that is going to tack care curtain things this mostly applicable for objective complex there class specification this on client and supplier something like caller of method we use so this is mostly applicable in object oriented so there is a term called client and supplier.

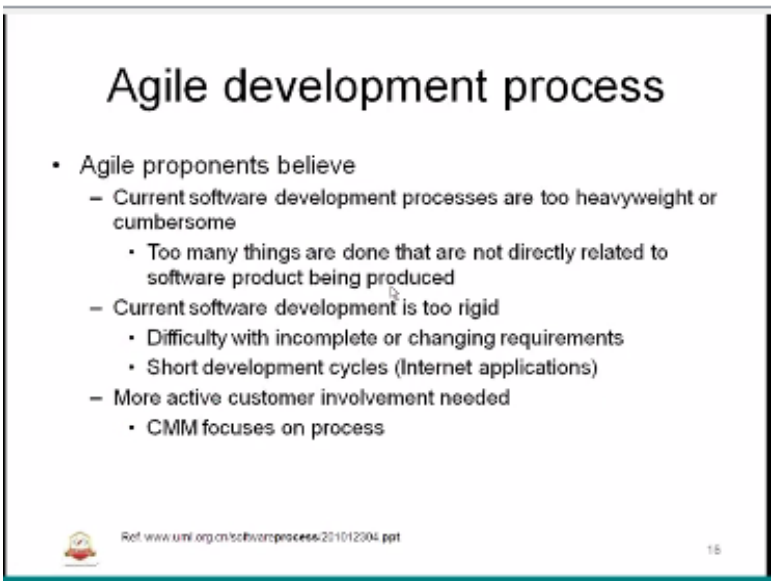
So the methods well be there the methods are emoted by calling function so the calling function is nothing but the client an d methods in a suppliers which supplies the column and approach that uses the decimations only to capture the design but also to encourage the interaction MM developers so basically throw documentations and models or any specification we are going to interrupt and with the help of that interrupt.

They are covering the testing in design by contract each module have interface specification that precisely describes what the module is supposed to do so the interface specification like face control document basally talks about what the module suppose to interface interact with the other sub modules or the main module.

So Meyer 1997 suggests that design by contract helps ensure that modules interoperate correctly this specification called a contract this two name it as contract is not a project contract something like that the name itself is contract so it calls the governs hoe the module is to interact with other module and system such specification cannot guarantee a modules or a correctness but it forms a clear and consistent basis soar testing and verification because we know that the interface what is suppose to be used accurse.

The entire system with the help of the specification so that we is the TMB the contract covers mutual obligation there reconditions benefits the post conditions and consistency constraints is called as invariants together contract properties are called assertions testers care about hoe this contract is enforced so how it is going to executed or enforced in that system on the test is what testers have to bother about so this is standard of the process that will be used in the testing this is basically used in the object oriental systems okay let us we come to another part of the embedded software tastings called as a agile testing or agile testing scrum process.

(Refer Slide Time: 51:56)



I will not talk in detail about this few slides we will try to go throw understand what is an agile development and what is that admin type of testing that is used in the industry okay so agile proponents believe that current software development processes are too heavyweight in many of industry they felloe typically the process what we have red out throw out this embedded software testing.

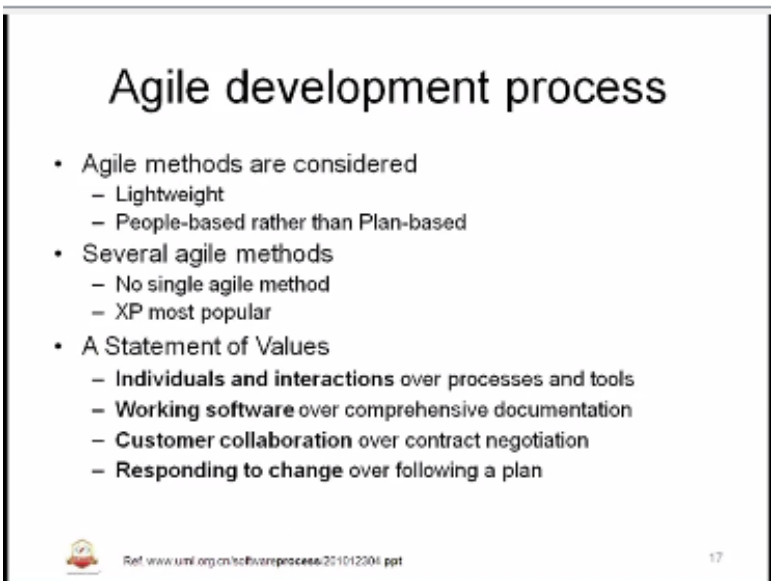
All this development process whatever it could be in terms formal planning design testing using varies methods un loge or any black box or white box check leafs in all this artery the part of the slandered process that we fellow but there are sudden industries like sense conductors all this definitely have process all that stuff but has a part of process what they called short term to market or time to market when the duration is less they think that this process is mostly useful or applicable.

So in that cases they use this so they believe that current software development process are to heavyweight or cumbersome So to many things are done that are not directly related software product being produced so eventually there are many thing like curtain check leafs guide lines for the sake of doing sometime we do though it is small piece of code we know that it is not going to crash or fail any case.

But we need to maintain because we are following that following the process for the entire thing does not matter how the big the complex how big the system is or how computed the system is or how small the unit is or whatever impact of the unit if it fails those cells sometimes it useful it is over barium short of thing.

So that is what agile proponents for their relive current software development is to relive physicality with the incomplete or changing requirements short developments cycles more actives costumers involvement is there CMM focuses on process we know that CMM capabilities in terms of CMM level capability model is called as up to five levels of maturity models are there like continue improvement all the stuff we also have a led stick sigma where we avoid this and all that all this are getting followed or used actively but they believe that we can improve on that curtain things so better to use that so what that called as agile process.

(Refer Slide Time: 55:18)



So agile process are methods are considered as lightweight process it is people based rather than plan based here it totally surrounding the process or the documentation or the check leafs or the guide lines or the under net process basically all the active are the methods that we use plan the people but in agile method we basically we will have people based or a team bases resource based rather than the process based or the plan based it a lightweight process several agile methods.

Exciting it is called a no single it is called as a XP there is another thing scram is model and extreme program in the XP that is also one of the model we have this all are basically agile methods a statements of values for the agile process is like individuals and interaction are technique that is over process and tools working software over comprehensive documentation.

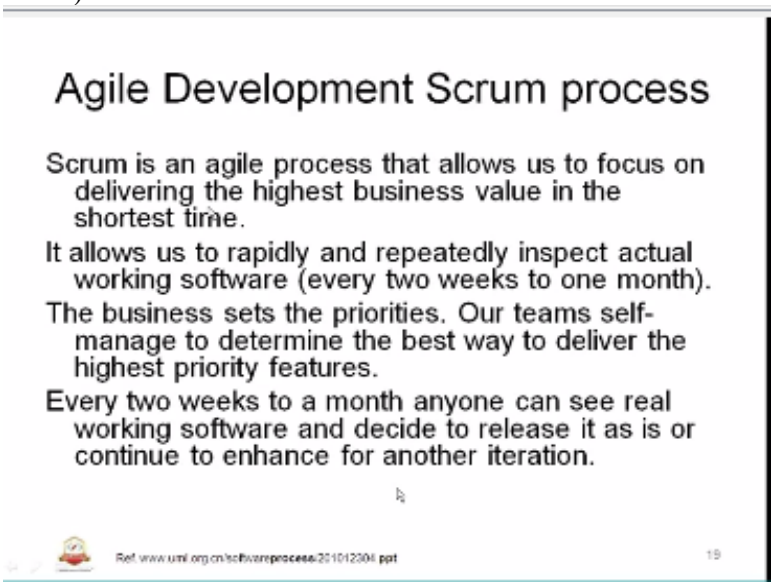
So that is what the values of agile process customer collaboration over contract negotiation so basically the need of the customer.

And customer client etc all the collaboratively looked in to in terms of agile developments and responding to change over following a plans as we fellow the plan we may have to adopt curtain changes into of the exciting plan so we no need to read it again all that following the plan with again short of plan we show that we can spent on this kind of change that we have for be done the agile are different agile methods are existing.

(Refer slide time: 57:52)

Scrum is one extreme programming is the other one adaptive software development ASD dynamic system development method DSDM like several methods there are been used scrum and extreme programming are most poplar it is also called as repute Porto type development rid there is a web site called as agile alliance they basically do resurge and try the promote this agile development that are we can go throw is that talks about agile development process in particular we try to understand agile development scrum process.

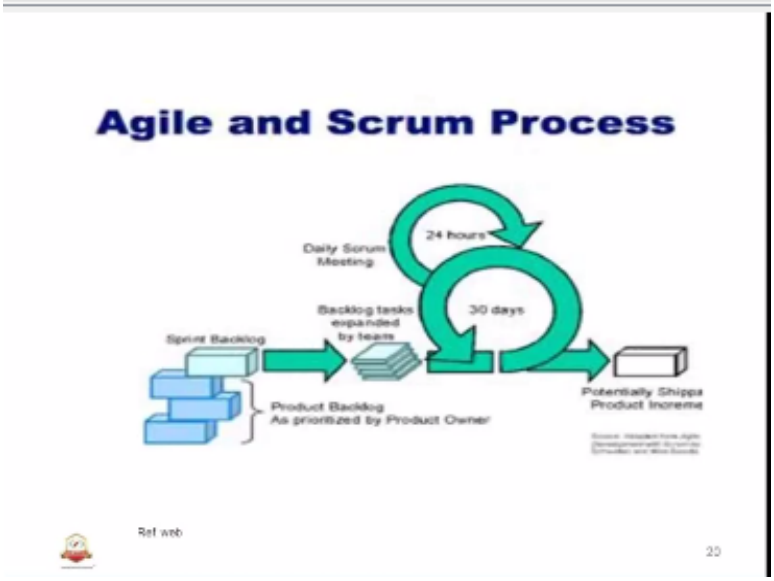
(Refer Side Time: 58:40)



Scrum is an agile process that allows us to focus on delivering the highest business value in the shortest time so basically it truly commercial short of a thing where time to market like highest business returns in shortest time that is going to be followed that is going to be used as process that is why it is called as scrum process it allows to rapidly and repeated inspect actual working software every two weeks to one month.

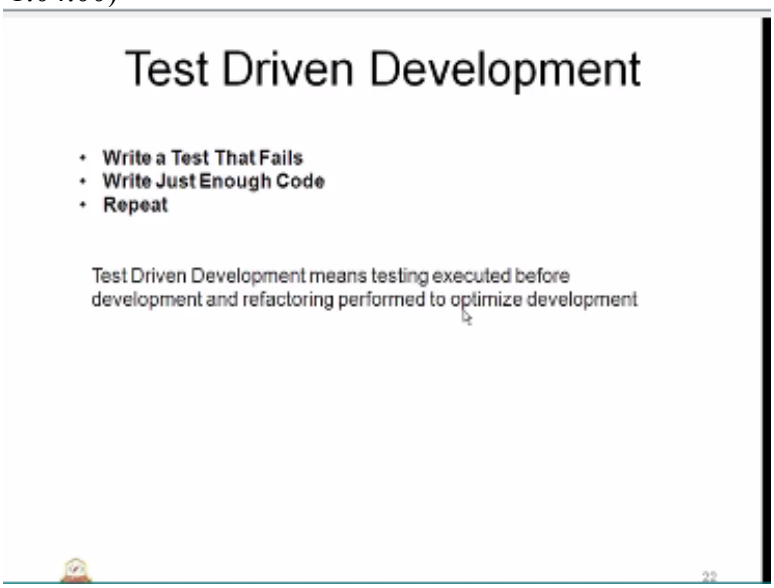
So during that frequency it could be reputedly inspected for the actual software or the working software the business sets the priorities our teams self manage to determine the best way to

deliver the highest priority features that means we have prototype that has to delivered to the costumer based on customer approval we are going to have the production of hours basically they are used in production this agile development scrum process.
Every two weeks to a month anyone can see real working software and decide to release it as is or continue to enhance for another iteration that is what we do in this scrum process.
(Refer Slide Time: 1:06:00)



You can see a diagram of debiting business scrum process that whatever the frequency that we had spoken here are they going to have the artifact or the developed items that are available so that is nothing but the sprint is a basically a check of varies actives so that tasks going to expanded by the agile process that is going to defined in 30 days timer and within the 30 days timer.
There are going to have another scrum meeting the scrum meeting well come up with different stack holders in the team and they well speck about what is going to happen today and so we are going to come with an interdict end of this scrum process or a check life of the defender.
So now coming to the testing process you may thing that testing is a needed because it is so short 24 hours 1 week or 2 weeks and all developments itself concuss maximum time so various scope of testing, testing can also fellow short of a quick scrum process or quick testing process is called agile testing scrum process so what are roles of tester in scrum? Actually there is no tester in formal scrum process so this the definition of tester only it part of the developing team only so testing is carried out by developer with unit test.
So he itself identify some segmentation try to stub drive the various units that are been used in the scrum so testing coverage testing is carried out by product owner / client basically dialing the client or the complete owner who ever has he will do the coverage what piece of software is working what is not working how much of functionality is been covered what are the requirements under gone like ways frequently testing by product owner each sprint that is back level of the various test testing acceptance criteria also be defined that is what we do argent

testing in scrum process mechanism in this type of testing that is called as a test drive development.
So basically what we do here is test development we execute before we actually we do a development on the developed part and we optimized the development as we progress as when we do a small checks of developments and we will try to test it so we basically test for the failures nature of the particular feature of the process test we have used so test driven development means testing executed before development of the computer development and refactoring perform to optimized development so again and again we will be tested.
So we write a test that fails write just enough code and repeat it this what we counting do it this is what is called as test driven development so basically it surrounds the test aspects of the particular unit and according we are going to have a development it is what is called as test driven development test management and control so this also important thing test control hoe are you going to have this test control.
(Refer Slide Time: 1:04:00)



So what to do when things happen that effects the test plans so you may have to revel the test plans that are effected something got change so what are those reallocation of resources changes to the test schedule changes to the test environment changes to the entry exit criteria changes to the number of test iterations changes to the test suite changes of the release data so this are the aspects of test plans how are you going to do it that is all with the control what we are going to do it.
So that is what is teat management and control so for ret management control also we will have a tools that is called as test management tools and that side one of my earlier session about this that is test link and bugzilla I will try to throw that quickly in the next class we will try to study about test management tool defect management tool etc so that consolation we studded about the test management, agile scrum test process test development agile process test design by contract and test process aliening to the software remodel so is that we will end the today session.