**[HVT][WS 6 Extra] Source Management part 2-20250507\_152327-Meeting Recording**

0:02  
So can we start?

0:04  
Yes, yeah.

0:06  
So Navin, actually yesterday we had a session related to the source code management and we tried to understand how the source code is getting managed.

0:21  
So I think most of the activity has been done at RNTCB level.

0:26  
So we would like to understand that process.

0:29  
So how you are managing the software related codes activities because there are some files or some requests come from other groups and your team process that and share the output to the other members.

0:45  
So that overall part we would like to understand.

0:50  
OK.

0:50  
So in terms of storage management, you mean?

0:54  
Yeah, storage management and from the software and system architecture point of view, any they mentioned that there is a specific like requests they shared with you for the validation purpose and your team kind of process that request and share the output of that particular file maybe for the software architecture related validations and all.

1:17  
So.

1:18  
So that part also would like to understand whether it is automated by software or any members from your team do and provide the output or how it works.

1:27  
OK, so first, when it comes to the storage management, when it comes to storage management, so I I think you might have seen the tools.

1:39  
So we use the kind of a tool, my M set.

1:42  
OK, my M set.

1:43  
So the of course, this is the kind of an user interface in the back end.

1:49  
The configuration management is the dimension.

1:52  
The tool is dimensions.

1:53  
OK, still the dimension.

1:56  
So the so we use we store them for each specification.

2:03  
So there is a version associated for for each code.

2:07  
So there is a naming conventions also has been followed for each item whatever we are going to store into the dimension.

2:17  
So Naveed, is it possible?

2:18  
Like if you show that tool and there if you explain that will be really helpful like level 2, because I can only store, I can only steer from the storage point of view.

2:31  
But next question, when it comes to the SYF architecture, I don't have the tool actually.

2:37  
But anyway, let's try to open at least on the configuration management one minute.

3:42  
Where is the.

5:00  
OK, let me share my screen.

5:02  
Yeah, sorry.

5:03  
You can I share my screen Stephen or somebody shared, OK.

5:21  
Let me know when you see my screen.

5:22  
Yeah, it's visible.

5:26  
OK, so this is the kind of an interface.

5:28  
OK, It's OK Naveen, can you speak a little bit louder?

5:32  
Your voice is coming from far.

5:35  
Hello.

5:36  
Is it better?

5:37  
Uh, no, I'm able to hear, but it's from far.

5:41  
Very bleak, actually.

5:45  
Is it better?

5:47  
Thank you.

5:47  
What is it?

5:49  
OK, OK, you can continue.

5:50  
Maybe if we do not understand, then we will ask you.

5:57  
Hello.

5:58  
Yeah, Yeah, please go ahead.

6:03  
Is it OK?

6:03  
Is it better?

6:04  
Yes.

6:05  
OK.

6:07  
So this is the kind of an user interface.

6:09  
What we use it in order to store the product in the back end.

6:13  
It will be interfaced to the dimension.

6:15  
OK.

6:15  
So we use I mean it is only the kind of a department, but of course we can configure it whatever the department.

6:24  
And then this is the kind of a OK, let check.

6:31  
And then so we it depends on the architecture of the codes.

6:36  
We will keep it here.

6:37  
Example EMS.

6:56  
So for the code item there is a type called B code.

7:06  
So here we can give a name whatever the name which we want to identify.

7:13  
And this is a kind of an create an SBM.

7:16  
So this is a this is for the SBM tool which is for each code item there will be a kind of a CDT which will be created kind of a unique number at so that it will you can see the flow in this tool that this code item has been stored and the the complete workflow will be closed.

7:44  
So then so here we will give the path which we are going to store and so this is the kind of a design part.

7:55  
So we will select particular part and this is a kind of an excel file which we need to fill it.

8:02  
So there is a template and then from the template we need to list all the items and associated version like that and then after that we need to load.

8:12  
So once we click load, then the items will be uploaded into the dimensions.

8:18  
OK, So here with the help of this structure, so you are actually storing the files into the dimension, right?

8:27  
Yeah.

8:28  
OK.

8:29  
So actually yesterday we had discussion about the variant management, so related to the software variant variants or version management, how it happens.

8:39  
So they explain or mention that they like create a first the configurations.

8:47  
So I think my set or that other tool is there where that configuration or the SOF creation is happening.

8:55  
So in SOF creation they are selecting the specific versions, the configuration and that way it is getting restored.

9:03  
But now they mentioned that like to downloading like fine they finalize that SOF.

9:09  
So once they hit, there is a files get downloaded from the dimension.

9:15  
So we would like to understand how if you manage those attributes or the variants or origin in the dimension.

9:21  
Also because that tool I don't have an access at first because because that is more from the project and also from the developer of the tool, we only will have an access, OK?

9:36  
Because this, so that tool is it's a my EMS, OK, what we call, yes, but of course that is also inside it is everything is interfaced to the dimensions finally in the back end.

9:51  
So only the name of the, the, the name of the tool and the user interface only it is a different, OK, But rest all, everything is connected to the dimensions, which is the configuration management tool.

10:04  
OK, So here you mean to say this my set or my means that other tool that is just UI created.

10:12  
Yeah, from dimension.

10:13  
So back end it is dimension only.

10:16  
Yes, yes, exactly.

10:17  
OK, OK, now, now that part is clear because we were having that confusion, OK with this tool, the configuration and all it is getting loaded, that selection happens and it is finally storing and fetching the information from dimension.

10:32  
So definitely from the dimension side there must be some script or some logic developed which will allow to pull required information.

10:41  
Yeah, exactly.

10:45  
OK, got it.

10:46  
So maybe let me flash my screen like we were having like listed some questions.

10:51  
So maybe from that angle maybe let us know which information you will able to share.

10:59  
Is my screen visible?

11:03  
Yeah, OK.

11:05  
So in the source code management topic, so mainly we discuss about that source code development process like how that software development happens.

11:17  
So our understanding is mainly in Matlab that model get created and based on that the source code get generated completely for application software and that source code is managed within a dimension.

11:35  
And this my M set or my Ms that those tools are mainly to have the kind of different like function feature that specification level configuration.

11:47  
So those details are managed through these tools.

11:51  
So that is what our understanding.

11:52  
So is there any other additional process or any additional input we would like to provide?

12:02  
No, your understanding is right, correct one.

12:05  
OK.

12:06  
And then here the source code manage, so source code manage repository is a dimension and for development and production it is the same.

12:16  
The dimension tool is getting used.

12:19  
So is that understanding also correct?

12:21  
Yes, exactly.

12:22  
OK.

12:23  
Yeah.

12:23  
The source code versions yeah that we discuss and traceability part, the specifically from source code to other object like task change.

12:38  
So in current scenario that is based on the attribute like you have the CR number or the issue number.

12:44  
So that issue number is just mentioned as a text, but it is not having direct link with SVM or that is not a case from source code binary subject code to other artifacts like task because we don't and the binary is object code.

13:02  
Mm hmm.

13:04  
Because we are only storing the C code the that's it.

13:09  
So that for once the final binary coming from the supplier like you provide application code and they like integrate with the base code and they provide a binaries for tuning purpose so that binaries are not getting stored in the dimension.

13:26  
No, we are not having the because we are only producing the C code.

13:30  
That's it.

13:31  
And after that the C code will be used by the ECU supplier and they will make the binary.

13:39  
OK, got it.

13:41  
And related to the source code branching.

13:43  
So the tools like you if you work in a gate SVN or bit bucket tools.

13:48  
So in normal software development process you can have the main branch and for the source and multiple people are working on the various parts.

13:57  
So that way you can create a different branches even for like branches for the release or integration and development.

14:06  
So that kind of different branches get created and you can integrate and have the combine the final source code.

14:12  
So I think the that based on the discussion, what we understood is so this feature of this functionality or the way of working it is not used done in a horse since one function is allocated to one team member and that person is completing the work and that through MATLAB that application the code is directly getting generated.

14:37  
So that branching related way of working, it is not done into the also is that understanding correct?

14:45  
Yeah, yes, yes, OK.

14:49  
And yeah, the code conflict I think, yeah, that is also not the case since one person is working.

14:56  
So this part is OK.

14:59  
So Sushanta, you were having one question in this, the source code partner for application level.

15:09  
Yeah, but Naveen is saying that only if the component dot C and dot edge is actually forwarding them towards suppliers, right?

15:21  
So, OK, so we got the answer.

15:24  
Yeah, yeah, I got the answer it the question is not arrived, Yeah, yeah.

15:34  
Then the software changes.

15:38  
So here what information we received that before finalising the software code of the source code if any changes needs to be done.

15:49  
So there is no CR or issue or bug get created.

15:52  
So it is managed at local level directly they modify the code and correct the code and perform the further testing and once it is a finalized and further shared to the supplier in that if any changes needs to be done then only kind of issue is is getting right.

16:13  
So can use as provide more insight on this based on our understanding.

16:25  
If you don't mind, can you repeat this point alone again?

16:28  
Yeah, yeah.

16:29  
So for software changes, so yeah, let's say if any change comes, what we understood is during the development, if any like that, they're also kind of a review is happening right at the code level or that.

16:44  
So there if any changes needs to be done then it can be done directly once the code is released or once code is released to the supplier and then if any changes needs to be done then it will handle through the issue.

17:00  
Yeah, yes, yeah exactly.

17:06  
So once the once the is is there are 22 things in fact.

17:12  
So if there is a so always the code is an associated to the specification version.

17:20  
And so whenever there is a specification changed, obviously the name of the code also will be changed.

17:29  
This is one case.

17:30  
There is another case is some cases we keep the same spec but there is a issue in the code inside the code.

17:39  
There is a bug in the code for that we keep the same spec version, but then the code version alone will change from 1.0 to 2.0 so that that's how it is managed.

17:52  
And then based on the whatever the specification they want then the respective code item will be taken.

18:03  
OK got it.

18:05  
And now the next question is related to the code review or the code quality.

18:11  
So once this the in a MATLAB that application architecture, the software is a get developed.

18:21  
I think the users are for mill testing users are creating some request.

18:25  
So that comes to your team, right?

18:27  
And then based on that if any changes or any calibration needs to be done, that input they received and accordingly they perform the changes.

18:39  
So can you explain on that particular part like process what input you received what exactly which tool you use and what output you provide to them?

18:53  
It is everything like I mean the specification version OK?

18:57  
The whenever when there is a new requirement then the respect to specification will be incremented from previous version like example from 2.0 to 3.0 OK it it is changed.

19:16  
So then once it is changed then this specification will be integrated into the SYF using a tool called My EMS My EMS tool and then the respective code will be taken into account.

19:38  
So that code like once that code is taken into account.

19:42  
So they, they, they mentioned that they create a request for validation purpose and that ticket will get created and accordingly they get the overall analysis of parameters and all.

19:55  
So that is done in RNTC.

19:58  
So that is what we got the details.

20:04  
Yeah.

20:05  
How the inside the tool is managed it?

20:07  
Yes it is managed like that, but the functionality is like this.

20:11  
I mean always.

20:14  
Like from the project, they will select which specification and the associated version they want from the project in order to create an SWF.

20:24  
That is, I mean from the project needs and based on the spec name and the version, the respective code item will be taken into account.

20:38  
OK, yeah, no that that that way I understood.

20:40  
But yeah, for that particular code I think before sending to supplier that partial validation that is happening, right?

20:48  
Yeah.

20:49  
So from that angle the users are creating some request in tool and they request that validation report.

20:57  
So that request I think coming to your team and they mentioned that email that OK, this request coming to team and the your team is sending them Excel file with the validation details.

21:08  
So the parameters or the values are correct or it need some calibration in that.

21:15  
And that is what they explained us.

21:19  
But they don't know exactly whether it is automated from the back end or someone from your team works on that because they said sometimes it takes a time 4 to 5 days or weeks.

21:33  
Till that time they without that final validation they need to send a code to the supplier and sometime it happened that some correction needs to be done and again they need to work on it and share the updated code to supplier.

21:47  
So that is what input we received.

21:52  
OK.

22:02  
So that part we just want to understand because they are also not sure which tool is getting used by your team for the validation whether it is a manual validation or it automated process.

22:18  
Because when it comes to the selection and maybe what what the project needs because only the selection of the item is manual but the rest all the process how to get it.

22:30  
It is everything is an automatic like what I have shown, just one case for the my M set the user interface.

22:37  
It is the same for my EMS.

22:39  
Also everything is an automatic which is an interfaced Naveen I.

22:46  
I think the question is related more to the coherence flux that is done on the spec and down to sorry Stephen, I didn't hear you.

23:10  
So I, I think the question is more related to the you know what Blagoi told us that when he needs to deliver a specific specifications, I don't remember the button in East, he pressed the button in East and then a request for auto condig is done.

23:34  
And then he said that there is a check behind and he received some emails about whether the variables are good or not.

23:48  
And also the he received the the code after auto coding.

23:54  
And he said that sometimes this process can take few hours, a few days or maybe weeks.

24:03  
Yeah, correct.

24:06  
But OK.

24:09  
Because when it comes to the auto coding, normally it is a fast until unless if there are any network issues, may be during the weekend, if it is stopped or may be the machine is switched off or something like that, then then only there will be a delay in the auto coding received.

24:26  
But otherwise otherwise it is a win must to receive.

24:35  
Now he gave us the example like the saturation block from the matlab.

24:41  
Like we are having the input which will have zero to 100 range and by multiplying with some multiplying factor and at the output side it will be more.

24:58  
The range will be more for the output variable, but somehow it is not captured in the MATLAB.

25:04  
So this kind of testing is actually doing in at, at at at at your side.

25:11  
And yeah, that will that will be captured from you and you will be intimating that within May.

25:21  
OK, yeah, yes, yes.

25:22  
But yeah, yeah, maybe there are 2 things.

25:25  
One is an auto coding certification and then there is another thing is on the design check kind of a checks.

25:33  
So, but this is also, but the time taking it is more more is because if if the people are requesting, all the people are requesting at the same time, the waiting, waiting period is high because the tool, the license which is used to check because there is only one license.

25:57  
And if the number of modules are higher which are in pipeline then it will increase.

26:03  
That's why some it it takes time.

26:06  
OK.

26:07  
So which tool it is you.

26:08  
So we want to understand that tool name.

26:10  
It is a BTC, OK, BTC tool.

26:13  
Yeah.

26:14  
And it is automated process or that pipeline?

26:18  
Yes, yes, it is a pipeline which is an automatic automation process.

26:21  
Yeah.

26:22  
OK, yeah, yeah.

26:23  
So that that part we want to understand so which tool is used for this process whether it is automated or it's manually done.

26:32  
OK, got it.

26:33  
So yeah, yeah that part and in a deployment process so this specifically not sure like I think this is not done your side but just want to confirm.

26:44  
So for building the source code, since you are sending application code to the supplier, so is there any building tool or is there any other areas where you are using any the build tool or continuous integration, continuous deployment.

26:59  
So per tools for any activity for horse projects, we, we are already using the continuous integration for the coding and back to back actually.

27:14  
So is it the same scenario that we just talk about from MATLAB to BTC or?

27:20  
Yeah, yeah, it is a similar actually we are using the CCD pipeline in order to test this, in order to use for the coding and the verification.

27:30  
So for CICD, which tool you use, we use a GitLab.

27:35  
OK.

27:39  
So for this, can you repeat that example like maybe that will help more.

27:47  
So for this CICD pipeline related activity, exactly what you exactly can you precede the question?

27:58  
I mean, yeah.

27:59  
So for CSD, you mentioned that GitLab is used, right?

28:02  
Yes, yes.

28:04  
Can you able to give one example like let's say in the MATLAB that architecture is get developed and then auto code generation happens.

28:14  
So where exactly you use at what step you use this CSD and GitLab?

28:19  
Yeah, because once the request has been made from the East, then there is a in, I mean automatically there is a request has been will be launched through in the Git lab and then then associated code generation all the steps because we have a steps like coding compilation and the static check will be performed.

28:48  
So this will be done through through CICD with the GitLab.

28:55  
OK.

28:56  
But GitLab is again repository, right?

28:58  
For CICD, Jenkins or any other tokens?

29:02  
Yeah, yeah, No, no, GitLab is the one which is used to make the pipeline.

29:07  
I mean to make the all this job creation to execute this pipeline GitLab is used OK.

29:21  
And further that code is again going back to the dimension like once that execution, yes.

29:30  
So we can say the MATLAB from MATLAB that request through and said that request will go through GitLab as a pipeline that execution, that code generation happens using that BTC tool.

29:45  
And from that BTC that generated code will come into the the dimension again.

29:53  
Yeah, exactly.

29:54  
OK, So Sushant, do you have a question or do you clear?

30:00  
No, I don't have this.

30:03  
OK, So BTC is used.

30:07  
Yes.

30:08  
OK.

30:08  
So GitLab is a pipeline through GitLab that pipeline process and BTC is actually the auto code, auto code generation is happening BTC tool, right Naveen, Yeah, yeah.

30:22  
OK.

30:23  
So it is used after the GitLab pipeline, right?

30:27  
Yeah.

30:30  
OK then.

30:31  
Yeah.

30:31  
Is there any interfaces, other interfaces also one we understood that GitLab is coming to the picture.

30:38  
So is there any other interfaces that is there with the BTC MATLAB?

30:47  
We we use another tool called tasking, which is the compiler actually, which one?

30:53  
Tasking, Tasking.

30:55  
OK, Yeah, yeah.

30:56  
Tasking you're using tasking as a compiler.

30:58  
OK.

31:04  
So which is the license tool?

31:09  
So compiling that code with the help of it OK, no BTC is the one which is for verification.

31:18  
OK Yeah, compilation is it is a task is a compiler.

31:23  
So once the code is generated, it will be compiled using the tasking compiler to ensure that there are no issues in terms of coding any no syntax errors like that.

31:39  
OK, got it.

31:40  
And are you performing that static code analysis?

31:47  
It is after like once the compilation is done and there is a static code analysis.

31:53  
And for this we use PC lit, but this is a but there is no specific license is needed for this.

32:01  
Which tool?

32:02  
PC lit, PC lit.

32:05  
OK.

32:06  
Can you put that into chat work that?

32:16  
OK, PC linked great.

32:20  
So that is for static code analysis and after that where the data will go.

32:27  
So tasking the compilation happens.

32:29  
Once compilation happens through PC linked that static code analysis has been done and then where the code will go because once because this is the end of the coding process, then after that the code will go to the dimension to store it.

32:49  
OK that it's in dimension.

32:51  
Yeah, OK, OK.

32:59  
And that is the one part and that auto star architecture generation.

33:06  
So you create a auto star architecture as well, right.

33:10  
So for that which tool is used, we use the MATLAB.

33:15  
We use the MATLAB only, yeah.

33:17  
But there is a specific toolbox inside the MATLAB auto serve blocks it you are saying?

33:24  
Yeah, yeah, yeah.

33:29  
How then how you are maintaining the composition?

33:36  
So the so when it comes to the code generation at each specification level, we use MAT works MATLAB product that is auto server toolbox to generate the code and err XML also.

33:51  
So this is one.

33:52  
But when it comes to the software, so software is nothing but it is a kind of an integration of several software components.

34:00  
So for this we create a composition file.

34:02  
But this composition, this composition is it is a kind of a internal script to merge all this err XML files.

34:14  
OK I someone is said our top, you are using our top to create auto SA architecture.

34:23  
So auto saree is Rtop is not for the making composition.

34:29  
Rtop is to to correct some of the issues for which MATLAB is not able to generate because MATLAB has several limitations.

34:41  
So this hotop is used to to to make it compliant.

34:45  
That's all.

34:46  
Yeah, do do do you have any like pain points in that?

34:54  
Because every component will create its own definition of the of the elements and it will have its own UID 's.

35:06  
So while merging may be it will get complicated for creating composition or system extract because UID it is not the big task because normally that I mean having a one having a same unique UID.

35:29  
Also the tools will not give any blocking.

35:33  
I mean it will give a kind of a warning, but no auto, no authoring tools will block the activity actually.

35:45  
OK.

35:51  
And yeah, OK, so the next related to the binary version handling.

36:00  
So today I think we had some discussion related to the that software release process that we were talking about.

36:09  
So there we understood that that final binary is like once do you supply or issue supplier combine that particular software that binaries were shared to this for calibration purpose calibration team and once they complete their calibration that binaries get stored in puke that from where the your plant takes that binaries to flash on the hardware.

36:41  
So the there are is your team involvement is there in that process, no deployment is handled.

36:53  
Which tool is because this binary handling it is a completely the supplier tools actually OK because each supplier have its own I mean own compiler or own tools to produce depends on the hardware depends on the project actually OK and but are you aware like who is normally responsible to flash the final binary is on to the ECU.

37:22  
So is it a supplier only or it is a the the flashing into the hardware is done by validation team.

37:29  
There is a specific validation team, OK, yeah.

37:32  
So some members from Hill, some members from the calibration team, it's like that yeah.

37:40  
So I'm talking about the final like actual your production flashing.

37:44  
So that handled by the validation team only right for each Yeah, yes, yes, exactly.

37:50  
OK, yeah.

37:53  
And I think this yeah, full so here specifically from this integration.

37:59  
So based on our discussion, we will just prepare one flow or the diagram connectivity so that tools that you mentioned.

38:06  
So we'll just try to plot that and maybe we'll share with you so you can your help is needed to validate and if any correction is needed in the data flow or that connectivity.

38:18  
So please help us in that area.

38:21  
Just just make sure that we understood correctly and the capture those detail properly.

38:26  
OK?

38:26  
OK, sure.

38:28  
Yeah.

38:28  
And I think from pain area point of view, I think the last time also you mentioned so that is not much pain areas from that for management part, right.

38:38  
So still saying that it is I I think it is mainly because of the maturity of the tools because as I said, because we are, we we started to make this tools.

38:53  
It's I mean nearly I mean in nearly 8 years, OK, nearly 8 years.

39:00  
So I'm not saying that we didn't have an issue before we had an issues, but now that it has been matured enough, that's why I said like we don't have a pain areas.

39:13  
I mean, OK, got it.

39:16  
Yeah, yeah.

39:17  
I think Sushant, any other question that we missed out or no?

39:28  
OK, Yeah, I think yes.

39:32  
So Panosh, Yusof, like Naveen, as we discuss, we'll map this tool connectivity and that we will share with you.

39:41  
So based on that, you please validate and provide your feedback if any correction or addition needs to be done.

39:48  
Yeah, of course.

39:48  
Yeah, sure.

39:52  
Yeah, I think we are.

39:52  
Good then.

39:53  
Yeah.

39:54  
Thanks for your support, Naveen.

39:56  
Definitely.

39:57  
We'll connect again.

39:58  
Yeah, Yeah, sure.

40:00  
Yeah.

40:00  
Thank you.

40:01  
Thank you all.

40:03  
Yeah, thank you, everybody.

40:04  
Have a nice thank you.

40:05  
Yeah.

40:05  
Bye.

40:07  
Bye.

40:07  
Bye.

40:08  
Bye.