



Review Report

Reporter:	Kenneth Skou Pedersen (ksp)			
Reviewer(s):	KKS, CSH, MVJ			
Reviewed Document:	Line coverage (User Guide)			
Document Number:	8434.521.03.001			
Review Date:	03 februar, 2003			
Comments:	Total:	Critical:	Major:	Minor:
	24	0	10	14
Duration:	30 minutes			
Conclusion:	Desk check after update			

This document contains a review report for a document under version control. Included are comments from the reviewers. This report is used as a part of the review procedure described in:

[TI 8352.230.02.001]

This report includes decisions for all issues raised by the reviewers. These decisions will be reflected in the changes made to the document under review.

Summary

Comment ID	Comment	Section	Page	Severity	Class	Decision	Consult
KKS001	O: As the type of document is a User Gui	-	-	Major	Unclear	A	
CSH004	First I would say that you have done a n	0	1	Major	Standard	A	
CSH011	Actually I think the title of the docume	1	1	Minor	Wrong	A	
CSH005	After my opinion a user_guide should not	2	5	Minor	Wrong	A	
MVJ002	O: But we would like it explained here..	2.2	5	Major	Missing	A	
CSH001	At this point I was missing how I should	3.1	8	Minor	Missing	A	
KKS002	O: This should be removed or only be a n	3.3	9	Major	Wrong	A	
CSH002	Actually this is not correct. In the cas	3.4	9	Minor	Wrong	A	
CSH003	You have already mentioned this in a pre	3.4	9	Minor	Unclear	A	
KKS003	O: In general I do not think that sectio	3.4	9	Major	Wrong	A	
KKS004	O: Again I think this is a bit irrelevant	3.5	9	Minor	Wrong	FR	PKR
CSH006	Which last line – I think a line is miss	3.5	10	Minor	Missing	A	
CSH009	Is it not possible to make a generic bat	3.5	10	Minor	Additional		
KKS005	O: Maybe an example with a PS entity ins	3.5	10	Minor	Unclear	A	
KKS006	O: I do not understand? What last line?	3.5	10	Major	Wrong	A	
MVJ001	O: The last line seems to have to do wit	3.5	10	Major	Missing	A	
KKS007	O: Again as this is a user guide this sh	3.6	10	Major	Unclear	A	
CSH010	This issue relates to CSH009. A more ge	3.7	11	Minor	Wrong	A	
KKS008	O: I think that section 3.7 should be mo	3.7	11	Major	Unclear	A	
CSH008	What will happen if profiling was enable	4	13	Minor	Unclear	A	
KKS009	O: It is not acceptable that the user sh	4	13	Major	Wrong	A	
CSH007	Where are these batch files located?	4	14	Minor	Missing	A	
MVJ003	O: The file "sm_cot.c" should be "sm_cof	4	14	Minor	Wrong	A	
KKS010	O: I do not think that the	5	17	Minor	Unclear	A	

Comment ID	Comment	Section	Page	Severity	Class	Decision	Consult
	appendix is n						

Comments

Reviewer: Carsten Schmidt		Comment ID: CSH001	
Page: 8	Section: 3.1	Class: Missing	Severity: Minor
Original text: The second job for the developer is to specify which source files are to be included in the profiling			
Comment: At this point I was missing how I should add the source files. I can first read this in section 3.5.			
Decision: A		Consult:	
Argumentation: This will be described better			

Reviewer: Carsten Schmidt		Comment ID: CSH002	
Page: 9	Section: 3.4	Class: Wrong	Severity: Minor
Original text: In the previous 2 approaches for performing the profiling was presented. Basically the results should be identical no matter which approach is chosen. The main difference will be in the implementation and execution time of the profiling.			
Comment: Actually this is not correct. In the case where the stack is be restarted for each testcase, you will have more hits on pei_create() functions and the content of these. But the xx_pei.c file could of course be left out of the profiling.			
Decision: A		Consult:	
Argumentation: Text will be corrected or removed if necessary			

Reviewer: Carsten Schmidt		Comment ID: CSH003	
Page: 9	Section: 3.4	Class: Unclear	Severity: Minor
Original text: As has been discussed in the previous it is not feasible to support only one of the approaches. This is due to the work that needs to be done in order to make all test cases able to run continuously. It will probably be much easier to implement a solution that supports both approaches than to clean up the existing code at the time. In addition to this, the test cases should be able to pass in both approaches.			
Comment: You have already mentioned this in a previous section -> remove the stuff. In general I think you could get rid of some stuff, since you already have mentioned it.			
Decision: A		Consult:	
Argumentation: This will be removed			

Reviewer: Carsten Schmidt		Comment ID: CSH004	
Page: 1	Section: 0	Class: Standard	Severity: Major
Original text:			
Comment: First I would say that you have done a nice job and put a lot of good work into the document. But there's a BUT :). I think that the document is too detailed - e.g. the users should read too much in order to be able to do line coverage. Maybe an initial section with a bullet list of things to do, files to use etc., in order to set up line coverage functionality, could be nice. Then you could "link" to more the detailed sections from the bullet points, in case the user wants more detailed info. Besides some information is out of interest from a user perspective.			
Decision: A		Consult:	
Argumentation: This will be corrected by dividing the document into two documents.			

Reviewer: Carsten Schmidt		Comment ID: CSH005	
Page: 5	Section: 2	Class: Wrong	Severity: Minor
Original text: For a design to pass full coverage, it is recommended that the line coverage for all modules in a design receive a very high coverage percentage. If a line of logic is not executed during simulation, the design has not been fully exercised. However it is not feasible to achieve 100 % coverage as this will take to much time compared to advantages. Even though a design does not receive 100 % coverage it is still not only a powerful tool for determining holes in the test suite it is also a method of documenting the level of test from time to time. That is when new test cases are added it is possible to measure and document whether ones new test cases actually tests untested code or already tested code			
Comment: After my opinion a user_guide should not state how high line coverage one should have, neither should it tell that it is not feasible to have a 100 % line coverage, because it takes to much time. This is up to project managers etc. to decide this ☺ I think this kind of stuff belongs in a "political" document and not a user guide.			
Decision: A		Consult:	
Argumentation: This will be removed. Perhaps this should be added to another document?			

Reviewer: Carsten Schmidt		Comment ID: CSH006	
Page: 10	Section: 3.5	Class: Missing	Severity: Minor
Original text: The last line is used to indicate that that it is the first run and it should be set to true for each time the profiling will be run. This information will be used by the batch job during run-time to determine when to merge the results			
Comment: Which last line – I think a line is missing?			
Decision: A		Consult:	
Argumentation: Paragraph is from older version and will be removed.			

Reviewer: Carsten Schmidt		Comment ID: CSH007	
Page: 14	Section: 4	Class: Missing	Severity: Minor
Original text: In order to start the actual measurements the setup_line_coverage.bat is executed. Following all the desired test cases are executed from the TAPcaller. Finally the post_processing.bat is executed.			
Comment: Where are these batch files located?			
Decision: A		Consult:	
Argumentation: This will be added. Has to be discussed where to place the files			

Reviewer: Carsten Schmidt		Comment ID: CSH008	
Page: 13	Section: 4	Class: Unclear	Severity: Minor
Original text:			
Comment: What will happen if profiling was enabled always in our msdev file? Then the developers can avoid checking g23_smi_umts.dsp file out and enabling profiling!			
Decision: A		Consult:	
Argumentation: This is already mentioned in the example with the SM entity. It will also be added earlier in the document.			

Reviewer: Carsten Schmidt		Comment ID: CSH009	
Page: 10	Section: 3.5	Class: Additional	Severity: Minor
Original text: The selection of which profiling method to perform will be selected by running a different batch job. This results in 3 different batch jobs. A typical configuration file will look like the following:			
Comment: Is it not possible to make a generic batch file, the developers can use for the UMTS project. I guess the file always would look the same, except the source files to be profiled? A possible idea is that the batch file could read the source files to profile from a list file? Doing this we could always maintain one batch file instead of maintaining one for each entity, in case of changes (switching to VS7).			
Decision:		Consult:	
Argumentation:			

Reviewer: Carsten Schmidt		Comment ID: CSH010	
Page: 11	Section: 3.7	Class: Wrong	Severity: Minor
Original text: This section will briefly describe the code/steps needed in order to perform the profiling in pseudo code. The details in the following are not intended for the developer but merely to clarify the structure for future updates performed by the tool group. This structure is common for all the profiling methods described above. Naturally there will be small differences in the arguments such as for instance the method used. However this is fairly simple and will not be described any further			
Comment: This issue relates to CSH009. A more generic approach could be nice, so that developers don't need to change batch files / settings in the project files every time they want to do coverage measurements. The only thing they should do is maintaining a file with a list of files to profiles. The required batch files should then be available from Clearcase. The file(s) should take a list file as argument. Therefore the information in section 3.7 is only relevant for tool group guys -> move it to a "design" document.			
Decision: A		Consult:	
Argumentation: This will be moved to a design document			

Reviewer:		Carsten Schmidt		Comment ID:		CSH011	
Page:	1	Section:	1	Class:	Wrong	Severity:	Minor
Original text:							
<p>Comment:</p> <p>Actually I think the title of the document is wrong, since the document also covers function and function time coverage. Maybe the "line" should be removed from the title.</p>							
Decision:				Consult:			
A							
Argumentation:							
Will be corrected							

Reviewer:		Kresten Kodal Sørensen		Comment ID:		KKS001	
Page:	-	Section:	-	Class:	Unclear	Severity:	Major
Original text:							
<p>Comment:</p> <p>O: As the type of document is a User Guide, I think that there is a bit too much explanations and descriptions of possibilities and what could be done instead of just stating the chosen facts.</p>							
Decision:				Consult:			
A							
Argumentation:							
The document will be divided into two documents namely a design document and a user guide.							

Reviewer:		Kresten Kodal Sørensen		Comment ID:		KKS002	
Page:	9	Section:	3.3	Class:	Wrong	Severity:	Major
Original text:							
<p>By a small enhancement the TAPcaller could support this approach by adding an option that allows the user to decide whether to restart the stack for each case. This way the profiling process would be more uniform no matter the approach chosen. The feature will be supported in the next release of the TAPcaller.</p>							
<p>Comment:</p> <p>O: This should be removed or only be a note. The User Guide should describe the situation as it is and not describe how it will be. If the situation change the User Guide must be updated.</p>							
Decision:				Consult:			
A							
Argumentation:							
This will be corrected							

Reviewer: Kresten Kodal Sørensen		Comment ID: KKS003	
Page: 9	Section: 3.4	Class: Wrong	Severity: Major
Original text: <p>In the previous 2 approaches for performing the profiling was presented. Basically the results should be identical no matter which approach is chosen. The main difference will be in the implementation and execution time of the profiling. As the first has to restart the stack for each test case and all of the results will have to be merged continuously this will increase execution time. However this is not expected to be a problem and is fully counterbalanced by the fact that this solution will fully support the existing code. As has been discussed in the previous it is not feasible to support only one of the approaches. This is due to the work that needs to be done in order to make all test cases able to run continuously. It will probably be much easier to implement a solution that supports both approaches than to clean up the existing code at the time. In addition to this, the test cases should be able to pass in both approaches. Even though the 2 approaches discussed in the previous differs much in the execution the implementation of them has a lot in common. The last approach is the easiest to implement but implementing the first will also support the last approach.</p> <p>As the current version of the TAPcaller does not support the feature to not restart the stack for each test case it is necessary to start the protocol stack manually when using this approach.</p> <p>Another important issue is how this profiling relates to Visual studio 7. Will it be possible to use the same tools or is it necessary to use some 3rd party tool? Both may result in changes to the batch job. According to the documentation for Visual studio 7 it should be possible to use but at least for time profiling better free tools exist. (TBD)</p>			
Comment: <p>O: In general I do not think that section 3.4 is needed.</p> <p>The first 3 paragraphs repeats already mentioned issues and the last 2 paragraphs describe a coming feature and potential future problems. Should not be in a User Guide.</p>			
Decision: A		Consult:	
Argumentation: This will be removed			

Reviewer: Kresten Kodal Sørensen		Comment ID: KKS004	
Page: 9	Section: 3.5	Class: Wrong	Severity: Minor
Original text: As a result of this it would also be recommendable to save the results from time to time. This saved information should include both the original format and the modified format presented in section 2.3. This way it will not only be possible to quickly get the coverage percentage but it will also be possible to use the information to perform work on the code.			
Comment: O: Again I think this is a bit irrelevant for the user guide. But it should of course be included in our development procedures.			
Decision: FR		Consult: PKR	
Argumentation: This will be removed. Perhaps this should be added to another document?			

Reviewer: Kresten Kodal Sørensen		Comment ID: KKS005	
Page: 10	Section: 3.5	Class: Unclear	Severity: Minor
Original text:			
Comment: O: Maybe an example with a PS entity instead of the test stack? Otherwise a reference to a template configuration file with a PS entity.			
Decision: A		Consult:	
Argumentation: This section will be corrected to reflect the UMTS example used later in the document.			

Reviewer:		Kresten Kodal Sørensen		Comment ID:		KKS006	
Page:	10	Section:	3.5	Class:	Wrong	Severity:	Major
Original text: The last line is used to indicate that that it is the first run and it should be set to true for each time the profiling will be run.							
Comment: O: I do not understand? What last line?							
Decision: A				Consult:			
Argumentation: Paragraph is from older version and will be removed.							

Reviewer:		Kresten Kodal Sørensen		Comment ID:		KKS007	
Page:	10	Section:	3.6	Class:	Unclear	Severity:	Major
Original text:							
Comment: O: Again as this is a user guide this should be e.g. a bullet pointed list of how to do a profiling measurement, instead of just a short repetition of the other sections!							
Decision: A				Consult:			
Argumentation: This will be corrected							

Reviewer:		Kresten Kodal Sørensen		Comment ID:		KKS008	
Page:	11	Section:	3.7	Class:	Unclear	Severity:	Major
Original text:							
Comment: O: I think that section 3.7 should be moved to an appendix, as it is not for the user.							
Decision: A				Consult:			
Argumentation: Will be moved to appendix or to another tool design document							

Reviewer: Kresten Kodal Sørensen		Comment ID: KKS009	
Page: 13	Section: 4	Class: Wrong	Severity: Major
Original text: In order to perform the line coverage measurements copy the following files to the same directory as the test file source files. (This will be changed so it will be possible to specify an output directory.)			
Comment: O: It is not acceptable that the user should copy files. There should be a set of common files stored a common place plus a set of files for each entity stored and released together with the entity. No copy and no local versions of files!!!			
Decision: A		Consult:	
Argumentation: Naturally this will be implemented.			

Reviewer: Kresten Kodal Sørensen		Comment ID: KKS010	
Page: 17	Section: 5	Class: Unclear	Severity: Minor
Original text:			
Comment: O: I do not think that the appendix is needed as the same information easily can be found in the Visual help.			
Decision: A		Consult:	
Argumentation: Will be moved to another document			

Reviewer: MVJ		Comment ID: MVJ001	
Page: 10	Section: 3.5	Class: Missing	Severity: Major
Original text: The last line is used to indicate that that it is the first run and it should be set to true for each time the profiling will be run.			
Comment: O: The last line seems to have to do with which source code files to include. Last line missing altogether?!?			
Decision: A		Consult:	
Argumentation: Paragraph is from older version and will be removed.			

Reviewer:		Comment ID:	
MVJ		MVJ002	
Page:	Section:	Class:	Severity:
5	2.2	Missing	Major
Original text: Basically it works like line coverage described above so it will not be explained further here.			
Comment: O: But we would like it explained here... :o) Seriously: Knowledge of whether a function is called/tested or not is vital to testing. Finding out that a function is not called is a much more serious sign of error than a single line. I know, I can see it from the list of lines, but a nicely formatted output of functions called, and more importantly, those NOT called, would ease testing considerably.			
Decision:		Consult:	
A			
Argumentation: The description of how to perform the Function coverage and function timing will be included into the document.			

Reviewer:		Comment ID:	
MVJ		MVJ003	
Page:	Section:	Class:	Severity:
14	4	Wrong	Minor
Original text: set srcfiles=/INC sm_cot.c(0-0) /INC sm_cop.c(0-0) /INC sm_cos.c(0-0) /INC sm_kef.c(0-0) /INC sm_kep.c(0-0) /INC sm_kes.c(0-0) /INC sm_tmf.c(0-0) /INC sm_tmp.c(0-0) /INC sm_tms.c(0-0) /INC sm_f.c(0-0) /INC sm_pei.c(0-0) /INC sm_qos.c(0-0) /INC sm_ttt.c(0-0)			
Comment: O: The file "sm_cot.c" should be "sm_cof.c", and the file "sm_ttt.c" should be "sm_tft.c".			
Decision:		Consult:	
A			
Argumentation: Will be corrected			