From Academia to Industry: Challenges and Lessons Learned from a Model-Based-Testing Experience

Emília Villani
Instituto Tecnológico de Aeronáutica
São José dos Campos – SP, Brazil
evillani@ita.br

Ana Maria Ambrosio
National Institute for Space Research
São José dos Campos – SP, Brazil
ana.ambrosio@.inpe.br

Thiago Dutra, Marcus V. O. Araújo
FIAT Betim, Brazil
thiago.dutra@fiat.com.br

Eliane Martins
Campinas State University - UNICAMP
Campinas – SP, Brazil
eliane@ic.unicamp.br

***Abstract*—This paper discusses the transfer process of a model-based-testing methodology from a scientific institute of the space sector to a vehicle production line from the automotive sector. It addresses the main challenges that were faced for the incorporation of the methodology in an industrial environment and summarizes the lessons learned from this practical experience.**

Keywords—model-based-testing; automotive embedded software; techonology transfer;

I. Introduction

In Brazil, each year around 160 PhD theses are defended in the area of Computer Science. It is a common belief that the contribution of most of these theses will remain restricted to the university or scientific environment, and will not be applied to the development of the Brazilian industry.

In order to contribute to the discussion of how to bring industry and university closer, this paper describes a practical experience of technology transference from the Academia to the Industry. The practical case is the introduction of CoFI (Conformance and Fault Injection) model-based-testing methodology in the production line of an automotive industry, with the purpose of validating automotive embedded systems developed by third parties.

Next section introduces the CoFI methodology and describes the organization of the technology transfer process adopted in the CoFI-FIAT Project. Following, we discuss the lessons learned and challenges tackled in the project.

II. The Practical Experience: CoFI-FIAT Project

The CoFI testing methodology, initially proposed in a doctoral thesis focused to the space sector, recommends test specification to be derived from state models of the system behaviour [1].

The testing team shall first identify the main services provided by the system under test and the related inputs and outputs that will be available for the testing. For each service, the team shall create a set of state models representing the system behaviour under four classes of inputs: (a) normal, (b) specified exceptions, (c) sneak path (when correct inputs arrive in unexpected instants), and (d) fault tolerance (when facing hardware faults). Then, these state machines are provided to the Condado tool [3], which generates the test sequences. Condado uses the switch cover method that visits the state machine transitions and produces a test suite. It is a transition tour algorithm that covers the combination of all reachable paths from the initial state. This process is illustrated in Figure 1.

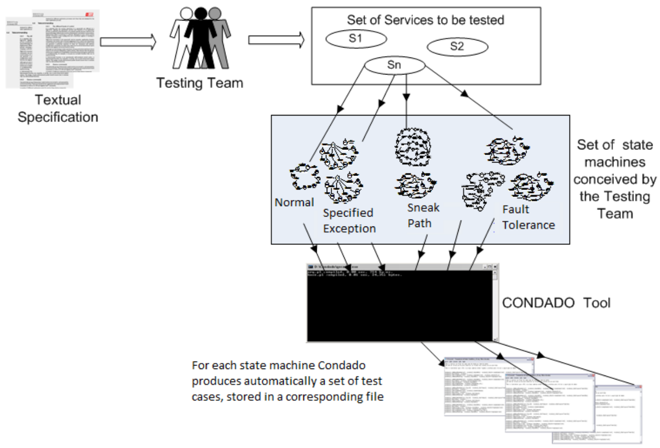

Figure 1. CoFI Testing methodology overview.

The purpose of CoFI methodology is to systematize the testing process as shown in [2] and [4], because of that, it attracted the interest of FIAT's Testing Team, which decided to invest in the establishment of a cooperation with the Academia. After preliminary contacts and technical discussions, a definitive meeting involving managers from INPE and FIAT, researches and technicians of all partners (INPE, UNICAMP, ITA and FIAT), defined the CoFI-FIAT project.

The CoFI-FIAT project consisted of introducing the CoFI methodology as part of the verification activities of the integration and test of the vehicle embedded systems.

Particularly, it aims at testing the functionalities implemented in the embedded software developed by third parties. These functionalities are known as Vehicle Functions (VF) and are described in a specification document that is the base for the application of the CoFI methodology.

The CoFI-FIAT transfer process was organized into two phases: (I) evaluation of the methodology concerning the modelling issues and the power to detect errors, and (II) introduction of the methodology in the development life cycle of the vehicle. While Phase I was achieved through the application of the methodology to 3 VFs of a vehicle that has already been developed, the purpose of Phase II is to apply it to 13 VFs of a car currently in the development line.

Phase I aimed at identifying limits, advantages and drawbacks of the methodology, as well as potential conflicts between the model-based-testing methodology and the current verification activities performed by the verification team at FIAT. The 3 VFs were attentively selected by FIAT team as the most critical and difficult to model. The project took seven months, as initially planned, and accomplished its initial purpose. The 3 VFs were modelled by an external team under the supervision of the researchers from the Academia. The modelling activity provided the identification of some problems in the VF specification, such as incompleteness and inconsistencies. Besides that, the way to create the state machines preconized by CoFI proved to be efficient for identifying scenarios propitious to errors. The results of Phase I are discussed in [5].

Because of generating automatically test case led to a huge number of test cases, automate test execution was mandatory. Consequently, Phase II had two main activities: to model all VFs in a time interval compatible with the development cycle, and to develop testing facilities to automatically execute the test cases in a time interval compatible with the development cycle.

These two activities were tackled simultaneously and both of them resulted in the identification of new challenges and the proposal of deep modifications in the development cycle. These challenges delayed the execution of Phase II and brought unforeseen and extensive improvements to the development cycle. Both challenges and improvements are discussed in the next section.

III. Phase II - Challenges and Lessons Learned

Figure 2 contextualizes the application of the CoFI methodology in the development cycle of a vehicle. CoFI modelling may start with the preliminary version of the Functional Design Requirements (FDR). Then, a HIL (hardware in the loop) testing environment should be developed. Both shall be ready when the FDR are frozen. When the prototype is received from the external producer (PROT), the test cases are generated and applied to it. The testing process should be finished by the end of the verification process (VP), which precedes the production of the pre-production vehicle (PS) and the start of the production line (SOP).

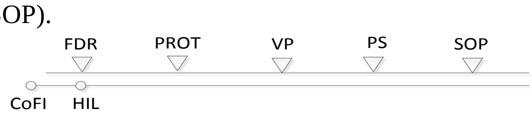

Figure 2. Development cycle and milestones.

A. Contributions to Documentation

Throughout the Phase II of the CoFI-FIAT transfer process, the team responsible for modelling the VFs as state machines had difficulties to understand the content of the VF and accomplish their work without the support from industry experts. Some of the frequent problems were:

- The team had difficulty to extract the external behaviour of the VF, i.e., the behaviour from the testing point of view. In some cases, the VF description mixed specification with implementation details.
- The description of the interaction of the VF with other VFs and external devices, such as sensors, actuators, etc. was confusing. In some cases, the access to the inputs and outputs of one VF had to be done through another VF, which may have run in the same component or not.
- The description of the VF's behaviour had ambiguities and/or was incomplete. The VF specification was provided in free textual form that varies according to its author.

These difficulties let to internal discussions at industry among the specification team and the testing team. As a result, modifications in the VF writing were proposed and a new VF model emerged. The new VF has the following features:

- The textual description of the VF is organized in a set of requirements, associated to an identification number, which is used for traceability purposes.
- The requirements are also associated with a set of use cases, described in terms of actors, actions and outputs. Which are used as an intermediate step in the process of modelling the VF in state machines, facilitating the communication among those people not familiar with state machine models.
- The VF has the list of other VFs that are affected by each of its requirement, for traceability purposes.
- It includes a table of input/output signals and corresponding hardware information, such as pin, relay and value. These input signals are related with the corresponding events of the state machine transitions. This list is important to concentrate all hardware information in a single point of the document, promoting the VF reuse.
- It includes a list of proxy parameters that affect the VF. These parameters are used to configure the vehicle embedded software according to the vehicle configuration.

The new VF requires more time and resources to be elaborated. However, the verification team expects that the benefits to the development process will compensate this cost avoiding rework at later stages of development.

B. Level of Refinement and Reuse

One key issue of the incorporation of CoFI methodology in the development cycle is the definition of the level of refinement of the state machine models. During Phase I, the level of refinement of the state machine models was defined according to the information available on the VF specification, and included hardware-dependent behaviour.

In Phase II, the team decided to adopt a high level of abstraction and did not include hardware-dependent behaviour in the state machine models. The hardware-dependent behaviour was coded in a separated software routine, called interpreter, that receives the high-level input events of the test cases and translates them into the appropriate physical signals to be sent to the testing environment.

In order to illustrate this issue, we use the fictional example of an input "passenger press a button" that causes the output "turn off the internal lights" (Figure 3). In the vehicle, this input is generated by an analogue button and it is recognized by the embedded system only when the passenger presses the button for at least 500 milliseconds. In Phase I, this input was modelled by two events, one associated with the rising edge in the analogue signal, and another with the falling edge, after 500 milliseconds. In Phase II, the same input was modelled as a single event that is received by the software interpreter. The interpreter translates it in an analogue input signal with rising and falling edges 500 milliseconds apart.

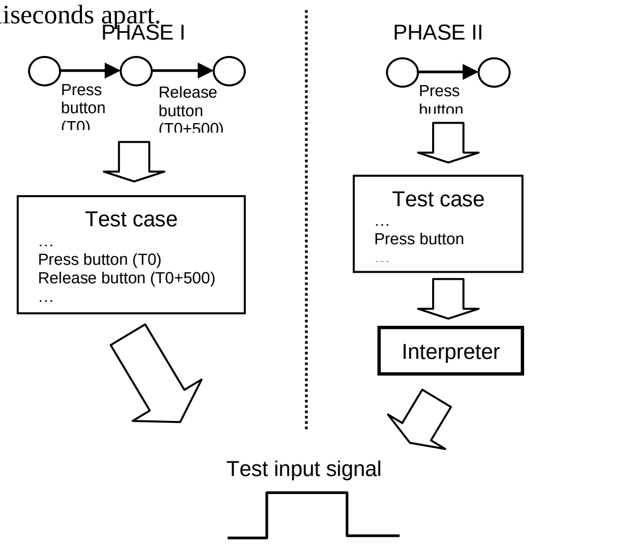

Figure 3. Different refinement level for state machine models.

The solution implemented in Phase II has two main advantages. The first and most important one is reuse. In the case that a new version of the car has a different interface, such as a digital button that generates a CAN message, the model and the test cases would still be valid. The second advantage is that this approach helps to manage the complexity of the state machine models, making them smaller, easier to develop, revise and understand.

The motivation for the approach adopted in Phase I is to have test cases where the button is released in less than 500 milliseconds. In Phase II, this kind of situation is tested separately, during the hardware equipment testing, and it was considered out of the scope of the CoFI methodology.

C. Development of Customized Software and Hardware Tools

One key issue about the transference process from academia to industry is the development of appropriate software and hardware tools. Usually the prototypes initially developed by academia for research purposes are not suitable for industrial environment. Among the common problems is robustness, capability of dealing with large amount of data. This issue is particularly sensitive if we consider that these tools do not have a market share yet. Their development has an associated risk that usually inhibits the interest of third companies to invest on it. As a result, the industry involved in the technology transference process has to choose between in-house development or contracting third parties.

In the case of the CoFI-FIAT transfer process, the industry option was for in-house development, both of hardware and software tools. This choice has the advantage of making possible the accommodation of new requirements during and after the development of the tools.

Concerning software, a new state machine editor was developed. Other than the user friendly interface, the main requirements for this tool that was not fulfilled by the corresponding academia tool is the automatic integration with other tools used in the process, such as Condado test case generator and DOORS, used for requirement management.

The integration among the tools aims at assuring traceability among requirements, state machine models, test cases, test execution models and test results. The new modelling tool assures that the events associated to transitions of the state machines models are the same events specified in the table of inputs/outputs of the new VF description. When the system under test fails in a test case, it is possible to identify which requirement has not been fulfilled. Similarly, when the requirements are modified, it is possible to know which models and test cases are affected.

Another challenge was the development of a testing environment that allows the automatic execution of the test cases. One initial solution was based on a controller and a set of modules from National Instruments. Due to requirements of compatibility with legacy systems, a second version was developed based on dSPACE solutions [6].

The testing environment includes the interpreter described in Section III.B. Basically, the interpreter should contain all the events described in the input/ouput table of the VF. It assures that the test case is independent of the vehicle hardware and testing environment. The same test case can be executed in the CompactRIO [7] or dSPACE solutions.

Another functionality implemented in the testing environment is the analysis of the output signals and the identification of inconsistencies (Figure 4).

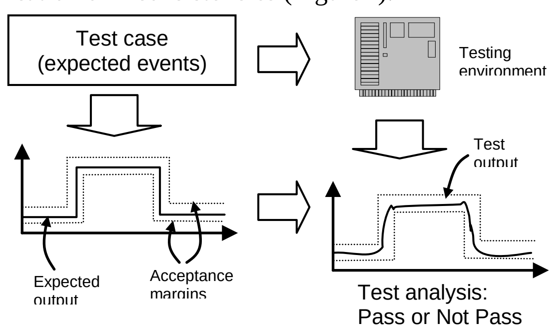

Figure 4. Testing analysis.

While the test case specifies the expected output as events, in practice, the output received by the testing environment is a set of signals in different formats, such as analogue signals and CAN messages. In order to analyse it, the interpreter receives the expected output from the test case and generates the corresponding physical signal. It then uses it as a reference to be compared with the real output. In the case of an analogue signal, an involucre is defined around the reference signal, with acceptance margins both in time and value. If the real signal is within the involucre than it is considered correct.

D. The testing team

The implementing of the CoFI methodology required the reorganization of the testing team at FIAT. A project team is now composed of four roles, defined as follows:

- Role 1 is the component master. He is responsible for the component test and validation. The component master must know which VF must be tested in order to validate the component and generates the service requests for the other 3 roles.
- Role 2 is the hardware developer. He is responsible for the configuration and development of the testing environment.
- Role 3 is the interpreter developer. He is responsible for developing the interpreter that receives the test cases and interfaces with the testing environment.
- Role 4 is the modeller. He is responsible for modelling the VFs and generates the test cases. He also performs the tests.

E. Contributions to Processes

The doubts of the modelling team working led the FIAT team to question the overall process of system design. They identified the lack of a system specification, and consequently, the lack of verification at system level.

Referring to the V development cycle (Figure 5), in practice, it means that verification was performed at component level and vehicle level. The system testing was usually performed at component level, and was dependent on the experience of the component tester.

This analysis led the FIAT team to propose a new broader definition of the vehicle function, at system level, the SVF (System Vehicle Function), which incorporates all the behaviour associated to functionality provided by the vehicle, and may include content of more than one VF, implemented in different components. The definition of the SVF is motivated by the difficulty of putting in the VF description all the necessary information to test it.

Associated with the creation of the SVF, a new verification level is also introduced in the development cycle, additionally to the component level.

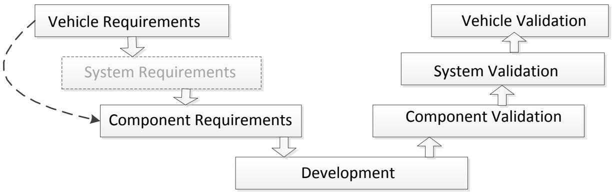

Figure 5. V development cycles.

IV. Conclusion

This paper discusses the transfer process of CoFI model-based-testing methodology from the Academia to the Industry. This process was initially structured into two phases: (I) evaluation of the methodology concerning the modelling issues and the ability to detect errors, and (II) introduction of the methodology in the development life cycle of the vehicle. The unexpected challenges faced during Phase II delayed the process but also brought unforeseen contributions to the industrial environment.

Some of these challenges were related to the need of introducing traceability, which required the improvement of existing processes and structuring/organizing the documentation. Adaptation was also required from the side of the Academia in order to clearly define an adequate level of abstraction that results in models with a manageable level of complexity. Another challenge was the development of customized software and hardware tools, and their integration with tools current at use - as academia tools were not suitable for the industrial environment, and no commercial tool was available, they had to be developed in house. Finally, a new organization of the verification team was defined to support all the new activities that resulted from the introduction of CoFI methodology.

From an industry perspective, future work will focus on determining the impact on budget and time schedule of the processes and solutions described in this paper. From the Academia perspective, future work will focus on defining strategies to select reduced sets of test cases based on the coverage of requirements.

Acknowledgment

Emilia Villani acknowledges CNPq (Process 306408/2014) for the financial support.

References

[1] A. M. Ambrosio, “CoFI – uma abordagem combinando teste de conformidade e injeção de falhas para validação de software em aplicações espaciais,” Tese de doutorado, INPE, São José dos Campos, 2005.

[2] Pontes, P. R.; Véras, P. C.; Ambrosio, A.M.; Villani, E. Contributions of Model Checking and CoFI Methodology to the Development of Space Embedded Software. Empirical Software Engineering – An International Journal, vol 10, No. 2. Springer. Online FirstTM, 11 July 2012.

[3] E. Martins, S.B. Sabião, A.M. Ambrosio, "ConData: a Tool for Automating Specification-based Test Case Generation for Communication Systems," Software Quality Journal, vol. 8, No.4, pp. 303-319, 1999. Kluwer Academic Publishers.

[4] A. M.Ambrosio, M. F Mattiello-Francisco, E. Martins, “An Independent Software Verification and Validation Process for Space Applications,” Proceedings of the 9th Conference on Space Operations (SpaceOps 2008). 12-16 May 2008, Heidelberg, Germany. American Institute of Aeronautics and Astronautics - AIAA, 2008.

[5] F. Mattiello-Francisco, E. Villani, E. Marings, T. Dutra, B. Coelho, A.M. Ambrosio. An experience on the technology transfer of CoFI methodology to automotive domain. In: 6th Latin-American Symposium on Dependable Computing (LADC)/Industrial Track, 2013, Rio de Janeiro. Proceedings of. Porto Alegre: SBC, 2013. v. 1. p. 1-4.

[6] https://www.dspace.com/en/inc/home.cfm

[7] http://www.ni.com/compactrio/