Technology Transfer in Computing Systems

D3.10: Individual TTP10 abstract

Project no.:	609491
Funding scheme:	Collaborative project
Start date of the project:	1st September 2013
Duration:	36 months
Work programme topic:	FP7-ICT-2013-10
Deliverable type:	Report
Deliverable reference number:	ICT-609491 / D3.10
WP and tasks contributing:	WP 3 / all
Due date:	30/06/2015
Actual submission date:	30/06/2015
Responsible Organization:	TUE
Dissemination Level:	Public
Revision:	1.0

TETRACOM D3.10: Verification In the Cloud TO Radically Improve Analyses

Wieger Wesselink, Tim Willemse (Eindhoven University of Technology), Robert Howe (Verum B.V., The Netherlands)

Formal verification of system designs helps reducing development costs by detecting issues early and by increasing the overall reliability of the system. Over a period of forty years, formal verification has grown from a pure academic exercise to a rich research area with promising industrial application. Many major firms have adopted these verification techniques in one way or another, using them to make better quality systems. However, the success of formal verification is directly linked to the maturity of the tooling performing the analysis and the skill of the verification engineer.

Most of the available tooling requires highly skilled and experienced verification engineers to tackle serious and complex industrial problems. In the past, Verum has created the ASD tool suite in an attempt to shield the system designer from the complexity of the verification language and technology by offering an intuitive integrated development environment for specifying complex, concurrent, industrial systems.

The ASD tool suite shields the complexity so well that it is easy to use for both novice and experienced system designers, but it limits the more experienced designers in accessing the full power of formal verification. Currently the ASD tool suite offers a proprietary design language and associated development methodology that is built on top of the verification technology offered by the FDR tool suite, which offers facilities for checking deadlock, livelock and refinement. As a first step towards making the ASD tool suite more appealing to experienced system designers, Verum has redesigned its modelling language, offering users more flexibility in modelling systems, and it has developed a novel tool suite called Dezyne. However, the verification technology is still built on top of the FDR tool suite.

In this TTP, we focus on developing the technology that allows Verum to use TU/e's mCRL2 tool set as the back-end technology for conducting verification, next to, or instead of FDR. This will permit Verum to offer new services to its expert users uncovering the associated added benefits; the new services that can be offered range from the ability to design data-dependent systems, to checking user-specific safety and liveness properties that go beyond deadlock and livelock, to offering advanced behavioural visualisation tooling to end-users. This will allow Verum to serve its customers better and will positively influence the quality of software at Verum's customers, setting a yardstick for software quality in general. The TTP allows the TU/e to extend their footprint and impact in industry through Verum's services. This in turn is expected to create a demand for improvements on theories and technology.

The software suite built and used by Verum is currently tightly integrated with the FDR tool suite. In order to couple mCRL2 to the Dezyne tool suite, in this TTP, we decoupled FDR and Dezyne by (partially) redesigning the software suite. The mCRL2 tool set is designed as a large collection (70+) of largely independent tools, but only a core of mCRL2 was needed to conduct the basic verifications offered by FDR. Within the TTP, we have been able to successfully connect the required mCRL2 core to Dezyne.

Experiments conducted at the end of the TTP, in which we compared the use of mCRL2 and FDR as a back-end show that the two are on a par, and for larger system models, the new mCRL2 back-end often outperforms the FDR back-end. While it is hard to tell (and beyond the scope of the TTP) whether this is due to differences in mCRL2 and FDR, or due to the connections of Dezyne and the back-end technology, these results are promising. Using the mCRL2 back-end, we have been able to successfully verify system properties that could not be verified using the FDR back-end. These results have inspired Verum to start investigating whether the technological advances offered by tools beyond the mCRL2 core can be integrated more tightly.