

NDT-Suite: A Model-Based Suite for the Application of NDT

Julián Alberto García-García¹, Manuel Alba Ortega¹,
Laura García-Borgoñón^{1,2}, and Maria Jose Escalona¹

¹IWT2 Research Group. University of Seville, Sevilla, Spain

²Aragon Technological Institute. Zaragoza, Spain.

{julian.garcia,manuel.alba}@iwt2.org,
laurag@ita.es, mjescalona@us.es

Abstract. In general, a methodology needs to be empowered by appropriate tool support. Despite MDE paradigm does not result friendly enough in enterprise environments, particularly, the application of transformations among models may become complex, monotonous and very expensive if there are no software tools automating the process. In this context, this research paper presents NDT-Suite. Nowadays, NDT-Suite is composed by a wide set of free Java tools which gives support to enterprises that are using NDT (Navigational Development Techniques) methodology in their projects. All of them support different aspects in NDT usage: quality assurance, exit generation or code checking, among others. These seeds set the environment for NDT usage for both research and practical use.

Keywords: Model-Driven Web Engineering, Model-Based Suite, Tools, Practical Experiences, NDT.

1 Introduction

The Model Driven Engineering paradigm (MDE) in general, and the Model-Driven Web Engineering (MDWE) in particular, came up in order to tackle the complexity of platforms and the inability of third generation languages to relief this complexity. This new paradigm intends to increase automation during the life cycle of software development and works, as primary form of expression, with definitions of models and transformation rules among these models by entailing the production of other models. In addition, if suitable tools are defined, this process could even be automatic.

However, MDWE is not easy to be applied in enterprise environments since it does not result too friendly for development teams. Concepts such as models, metamodels, transformations or QVT, among others, are not common notations in the enterprise environment and they seem too abstract and complex.

For this reason, this research paper presents how NDT [1] (Navigational Development Techniques) addresses this challenge with the aim of involving the enterprise with the power of MDE. NDT is a methodological proposal included within MDE that provides support to all phases of software life cycle: feasibility study,

requirements, analysis, design, implementation, testing and maintenance phases. For each development phase, NDT defines a set of metamodels and proposes a set of QVT transformations that enables to get one phase results from the previous one.

This paper is structured as follows. After this introduction, Section 2 presents the suite of tools for NDT. Finally, in Section 3 the final conclusions are expounded.

2 NDT-Suite

NDT-Suite¹ is a set of free Java tools that facilitates the application of NDT in real projects. With this suite, enterprises can benefit from the advantages of using MDE in their projects.

NDT-Suite works on/with a UML- based tool named Enterprise Architect² (EA). To select Enterprise Architect did not result an easy task. In fact, a comparative study developed by our research group and the Andalusian Regional Government concluded that this was the tool that offered the best value for money. Furthermore, EA offers several important advantages, such as the possibility of defining profiles or tools for document management by drawing UML diagrams, for instance, which have been very relevant to carry out our work.

Currently, the suite of NDT is composed by the following tools:

- **NDT-Profile:** it is the main tool for NDT usage. This tool is composed by a set of UML-profiles which were developed for each metamodel of NDT. These UML-profiles were defined in Enterprise Architect. With this, NDT-Profile offers the chance of gathering all the artifacts that define NDT easily and quickly, as they are integrated within the tool Enterprise Architect. Figure 1 shows a perspective of NDT-Profile.. Area (a) shows a diagram associated with the information storage requirements. Area (b) shows the package where these requirements are stored. With NDT-Profile, all NDT artifacts can be graphically specified. Area (c) shows the toolbox related to these requirements. It offers the possibility of collecting all the artifacts that define NDT in an easy and quick manner.
- **NDT-Driver:** it is one of the main tools of NDT methodology. It is completely based on NDT-Profile and implements a set of automated procedures so as to carry out each of the QVT transformations defined in NDT. It generates the analysis models from requirements, the design models from the analysis and the tests models from requirements. In addition, NDT-Driver allows obtaining the model requirements from the requirements collected within the feasibility study phase of the project. Moreover, NDT-Driver can be used in projects using both, a sequential life cycle and an evolutionary life cycle. Once transformations to perform have been selected, models to generate can be chosen.
- **NDT-Quality [2]:** it is a tool that automates most of the methodological review of a project developed with NDT-Profile. It checks both, the quality of using NDT methodology in each phase of software life cycle and the quality of traceability of MDE rules of NDT. It also provides a report in different formats describing the inconsistencies appeared during the review.

¹ NDT-Suite available from <http://www.iwt2.org>

² Enterprise Architect available from <http://www.sparxsystems.com>

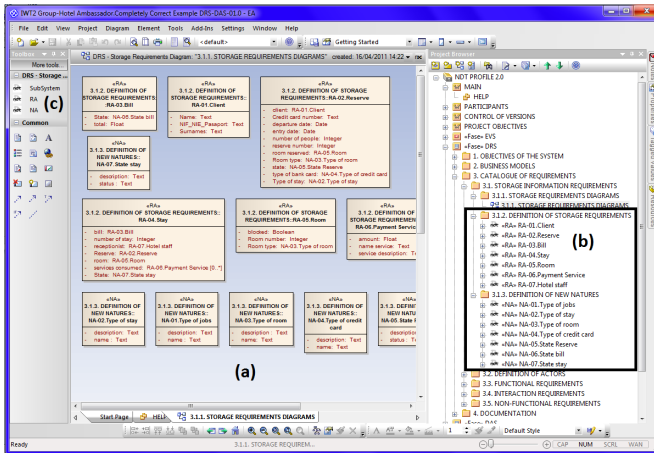


Fig. 1. A NDT-Profile Perspective

- **NDT-Prototypes:** it is a tool that generates a set of XHTML prototypes from the navigation models described in the analysis phase in a project developed with NDT-Profile. This tool is not related to the test phase, although it gives very good support for requirements validation.
- **NDT-Glossary:** it is a tool in its suite offered by NDT which uses the model-driven paradigm to generate a glossary from the requirements model. This tool allows engineers to gather and define the more relevant and critical concepts to the system. Furthermore, the use of a common glossary reduces the risk of misunderstandings and facilitates communication between users and analysts.
- **NDT-Counter:** This tool provides a measure of effort required to develop a project. This measurement is based on the use case technique [3].
- **NDT-Report:** it is a tool to generate PDF documents from NDT-Profile.

The demo of [8] shows how to use NDT-Quality to validate the requirements phase which has been specified using NDT-Profile, how to use NDT-Driver for generating the different models of the analysis phase the from requirements phase, and finally how to use NDT-Prototypes to generate XHTML prototypes.

3 Conclusions

The application of MDE becomes complex, monotonous and very expensive if there are no software tools automating the process. To meet this need, NDT has defined a set of supporting tools called NDT-Suite. In the last ten years, NDT and NDT-Suite were used in a high number of real projects. In fact, NDT-Suite is currently being used in several projects developed by different companies, either public or private, big or small. On the one hand, public companies such as the Andalusian Regional Cultural Ministry, the Andalusian Regional Health Ministry, among others, are using NDT and NDT-Suite. Private IT companies in Andalusia also are using NDT in some of their projects.

The main advantage of NDT-Suite is that it reduces the cost of ensuring the quality and traceability of the deliverables carried out during the project development phases (requirements, analysis, design, etc.). It also reduces the cost of deliverables from one phase because they are obtained from other deliverables of the previous phase, through MDE rules. On the other hand, we are working on how to extend NDT to provide support in the implementation phase of a Web project. Only few Web engineering methods support the systematic development of Web applications with a mature CASE tool. Thus, some methodologies like OO-H methods [4] with VisualWADE³ and WebML [5], WebRatio⁴, OOWS[6] with Olivenova⁵ or UWE[7] with UWE4JSF⁶ are some specific solution that offer code generation in a MDE context.

Acknowledgement. This research has been supported by Tempros project (TIN2010-20057-C03-02) and Red CaSA (TIN 2010-12312-E) of the Ministerio de Ciencia e Innovación, Spain and NDTQ-Framework project of the Junta de Andalucía, Spain (TIC-5789).

References

1. Escalona, M.J., Aragón, G.: NDT. A Model-driven Approach for Web requirements. IEEE Transaction on Software Engineering 34(3) (2008)
2. Escalona, M.J., Gutiérrez, J.J., Pérez-Pérez, M., Molina, A., Martínez-Force, E., Domínguez-Mayo, F.J.: Measuring the Quality of Model-Driven Projects with NDT-Quality. In: Information System Development, vol. 1, ch. 26, pp. 307–317. Science Business Media, LLC 2009, USA (2011) ISBN/ISSN: 978-1-4419-7355-9
3. Karner, G.: Resource Estimation for Objectory Projects. Objective Systems SF AB (1993)
4. Gómez, J., Cachero, C., Pastor, O.: On Conceptual Modeling of Device-Independent Web Applications: Towards a Web-Engineering Approach. IEEE Multimedia 8(2), 26–39 (2001)
5. Ceri, S., et al.: Designing Data-Intensive Web Applications. Morgan Kaufmann, San Francisco (2002)
6. Fons, J., Pelechano, V., Albert, M., Pastor, Ó.: Development of Web Applications from Web Enhanced Conceptual Schemas. In: Song, I.-Y., Liddle, S.W., Ling, T.-W., Scheuermann, P. (eds.) ER 2003. LNCS, vol. 2813, pp. 232–245. Springer, Heidelberg (2003)
7. Koch, N.: Software Engineering for Adaptive Hypermedia Applications. Ph. Thesis, FAST Reihe Softwaretechnik, vol. (12). Uni-Druck Publishing Company, Munich, Germany (2001)
8. IWT2. Video demonstration of NDT-Suite available from canal youtube of IWT2 Research Group,
<http://www.youtube.com/watch?v=uLDrqz9t690&feature=plcp>

³ <http://www.visualwade.com/>

⁴ <http://www.webratio.com>

⁵ <http://www.care-t.com>

⁶ <http://uwe.pst.ifi.lmu.de/toolUWE4JSF.html>