Name of Authors:	Bruno Legeard – Jean-Pierre Schoch
E-mail of Contact Person:	legeard@smartesting.com
Telephone Number:	+33 687585869
Contact Address:	48 rue de la Bienfaisance – 75008 Paris - France
Tutorial Title:	**Model-driven testing for Information systems – From business requirements to test repositories**
Intended Audience:	Test engineers, Project Managers, QA Engineers

Tutorial Overview

This tutorial shows how model-driven testing renews the whole process of functional software testing for IT projects. Model-driven testing provides full support for test production and maintenance, with manual and/or automated test execution. It supports the phases of designing the test generation models, generating and documenting the tests, producing and maintaining the bi-directional traceability between tests and business requirements, and implementing the test automation layer.

This tutorial addresses these points by presenting a tool-based model-driven testing for Information Systems process, giving the specifics of testing large-scale information systems, including packaged applications (ERP, CRM, HRM, …).

From this tutorial, you will learn:

- How to use business process models and behavioral models for automated test generation for Information systems
- How to automate bi-directional traceability between tests and business requirements
- How model-driven testing connects with automated test execution platforms
- How to integrate model-driven testing in the testing process (typically ISTQB testing process)
- The benefits and limitations of model-driven testing for IT projects, its cost effectiveness and how it can reduce time-to-market

This tutorial takes a practical and easy-to-understand approach to model-driven testing, using Smartesting CertifyIt test generation tool.

Tutorial Structure and Timing

<u>Half-day tutorial</u>

Structure:
- Current challenges of testing large-scale information systems
- Levels of testing addressed by model-driven testing for information systems
- Composing Business Process models, Behavioral models and Test data models for automated test generation
- Working example: OrangeHRM (packaged applications)
- Integration into the IT testing process
- Summary – Conditions for deployment and best practices

Tutorial Materials (Handout, Demo, etc)
- o Presentation and Demo

Evaluation Criteria

1. What aspect(s) of model-driven testing is the focus of this tutorial?
(modelling, test generation, deployment, integration of generated tests, test process integration, etc)

- *Specifics of model-driven testing in the context of IT projects*
- *Modeling aspects (Business Process models, Behavioral models, Test data models)*
- *Test Process integration*

2. What is the level of model-driven testing expertise required by the audience to follow your tutorial? (Beginner, Intermediate, Expert)

Intermediate

Further Details: Authors presentation

Bruno Legeard, CTO of Smartesting and Professor of Software Engineering at The University of Franche-Comté (France).

Prof. Dr. Bruno Legeard, co-founder and CTO of Smartesting, is internationally recognized as an expert and a well known speaker in the model-driven testing field. He has given keynotes at numerous testing and software engineering conferences.

He is experienced in deploying model-driven testing solutions both in enterprise information systems area and in the embedded systems field. In 2007, Bruno Legeard authored with Dr. Mark Utting the first industry-oriented book on model-driven testing, "Practical Model-based Testing: A Tools Approach", Morgan & Kaufmann Publisher.

Bruno is a member of Program Committees of several major software testing international conferences such as ICST – International Conference on Software Testing, Iqnite and MVV – Model-driven Validation and Verification.

Jean- Pierre Schoch, Model-driven testing Expert and Chief Architect at Smartesting

Jean-Pierre joined Smartesting in May 2008 after 13 years with Rational in the United States and in France promoting object-oriented technologies, and more specifically Model-Driven Development and Model-Driven Architecture approaches. The decision to join Smartesting was largely motivated by the prospect of contributing to the development of a Model-driven Testing approach to industrialize software testing.