

# Property-Based Testing - The ProTest Project

John Derrick<sup>1</sup>, Neil Walkinshaw<sup>1</sup>, Thomas Arts<sup>2</sup>, Clara Benac Earle<sup>4</sup>,  
Francesco Cesarini<sup>3</sup>, Lars-Ake Fredlund<sup>4</sup>, Victor Gulias<sup>5</sup>,  
John Hughes<sup>6</sup>, and Simon Thompson<sup>7</sup>

<sup>1</sup> Department of Computing, University of Sheffield, Sheffield, S1 4DP, UK

<sup>2</sup> Goteborgs Universitet, Goeteboug, Sweden

<sup>3</sup> Erlang Solutions Ltd, London

<sup>4</sup> Universidad Politecnica de Madrid, Madrid, Spain

<sup>5</sup> Lambdastream Servicios Interactivos SL, A Coruna, Spain

<sup>6</sup> Quviq AB, Svedalen, Sweden

<sup>7</sup> University of Canterbury, Canterbury, Kent, UK

[J.Derrick@dc.shef.ac.uk](mailto:J.Derrick@dc.shef.ac.uk)

**Abstract.** The ProTest project is an FP7 STREP on property based testing. The purpose of the project is to develop software engineering approaches to improve reliability of service-oriented networks; support fault-finding and diagnosis based on specified properties of the system. And to do so we will build automated tools that will generate and run tests, monitor execution at run-time, and log events for analysis.

The Erlang / Open Telecom Platform has been chosen as our initial implementation vehicle due to its robustness and reliability within the telecoms sector. It is noted for its success in the ATM telecoms switches by Ericsson, one of the project partners, as well as for multiple other uses such as in facebook, yahoo etc. In this paper we provide an overview of the project goals, as well as detailing initial progress in developing property based testing techniques and tools for the concurrent functional programming language Erlang.

## 1 Introduction

Communication networks, based on telephony, wireless and Internet, have over the last few years been converging. At the present time and for the foreseeable future, more and more services will be added to these merging networks. Moreover, these services are becoming more complex, both in themselves and in their interactions with each other and their end users. The telecoms industry has an admirable record in providing reliability and robust services to its clients, and indeed it is the telecoms industry that can point to 5-nines reliability: that is 99.999% reliability, of their core systems.

This context provides the motivation of the ProTest project - namely that of maintaining 5-nines reliability in future service-oriented networks and systems.

The software for new services and network devices is rapidly growing in complexity, among other things because of the variety of formats and multiplicity of delivery modes evident in modern communication protocols (with thousands

of optional fields, for instance). In addition, such software needs to be context-aware, since the requirements vary when the same software is used in different ways. There are several ingredients for ensuring that such complex systems provide the expected reliability, among them choosing a good architecture, using the right technologies, improving the software process, and also being extremely thorough and efficient in *testing*.

Testing of complex systems is difficult and time-consuming in the extreme, and in the ProTest project we build upon the innovative idea of using properties as objects for testing software. In order to deliver dynamic services and interoperable network applications with guaranteed properties, we focus testing around these properties.

The economic motivator is that testing with properties as objects improves the competitiveness of software developers, since they can deliver higher quality software for a lower price. It also allows collaborating companies to improve the definition of their software interfaces and therewith improve the compatibility between their services.

Our objective is to deliver methods and tools to support property-based development of systems, and in order to do so we need tools to integrate property-based testing into the development life cycle. To this extent we are conducting work along four technical themes as follows:

*Property discovery.* Current testing is based on sets of test cases embedded in test suites; over the lifetime of the project we will aim to provide tools to aid the software developers to extract properties from this test data. Current specifications and models are often informal: so we will develop specialised property languages to ease the formalisation of existing specifications.

*Test and property evolution.* All software systems are subject to change and evolution; we will thus provide tools to support the evolution of tests and properties in line with the evolution of the system itself.

*Property monitoring.* Not all properties can be tested in advance of systems being executed, and so we will provide tools to support the post hoc examination of trace details for conformance to (or indeed violation of) particular constraints.

*Analysing concurrent systems.* At the heart of service oriented systems is concurrency: servers will provide services to multiple clients in a seamlessly concurrent way; services will federate to provide complex functionality through concurrently performing parts of a task. We will provide tools by which such concurrent systems can be analysed for fundamental properties by way of model-checking and testing.

In subsequent sections of this paper we explain work in progress under each of these themes.

## 2 Background

The ProTest project aims to introduce property-driven development into the software engineering process. Property-driven development can be used in a variety of programming languages and systems. The particular platform chosen for