# Model based approach to verification of higher-order programs

## (Invited Talk)

Igor Walukiewicz
CNRS Bordeaux University, France
igw@labri.fr

Higher-order programs manipulate not just data, as first-order programs do, but also other programs. Higher-order features make code shorter, more structured, and more modular. They offer mechanisms to deal with such fundamental issues as scalability in multi-processor architectures, or fault-tolerance. In this talk we will present some recent advances in verification of of both safety and liveness properties of higher-order programs.

To analyse liveness proprties, we need to consider infinite behaviours of functional programs. We regard programs as constructing a possibly infinite execution tree that represents the order in which atomic operations are executed. This approach faithfully models the control flow of a program, while abstracting from its data part. We then use monadic second-order logic, or equivalently parity automata, to express properties of execution trees. We discuss in what sense this approach encompasses many known program analyses, and how it can be also used for program transformation.

Submitted to:
VPT 2016