MOTIVE: Method for Object Testing, Integration and Verification

We describe an approach to specifying and testing object-oriented and component-based systems. Specifications are constructed from Object Machines, an adaptation of the successful X-Machine. Fine detail is added using an Object Algebra. Both machines and algebra are subject to refinement by rule, which corresponds to realisation or to specialisation in the object-oriented model. Rules for composition allow individual objects to be completely tested under Chow-style assumptions, and integrated systems to be composed, such that the integration object may be tested in isolation, reducing the cost of testing overall.

In this talk, we explore what we do when we derive tests from a Z specification, and compare this process with deriving tests from a specification written in a process algebra. We suggest a framework by which these views might be unified.

The problem of testing from an extended finite state machine (EFSM) is complicated by the presence of infeasible paths. This talk considers the problem of expanding an EFSM in order to bypass, or limit, the infeasible path problem. The approach is developed for the specification language SDL but, in order to aid generality, the rewriting process is broken down into two phases: producing a normal form EFSM (NF-EFSM) from an SDL specification and then expanding this NF-EFSM.