Improving the Specification of Data Types in LOTOS

Doctorate thesis

Charles PECHEUR, University of Liège (B)

Extended abstract

The general goal of this dissertation is to provide solutions for making the specification of data types in LOTOS more concise, easier to write and understand and less dependent on environment constraints. This task is considered mainly in the framework of the existing LOTOS standard, but also in the perspective of a revised LOTOS. A generic text translation formalism is defined in support of the former case.

LOTOS is a specification language used for describing distributed systems. LOTOS describes data types abstractly in the language ACT ONE, which uses conditional equations and has algebraic semantics. While this model is very abstract and general, its fundamental nature leads to very lengthy and technically complex specifications. Moreover, equational specifications are not computable in general, so existing software tools use more efficient evaluation solutions that work correctly only for a restricted class of specifications. Data types are recognized as a major weakness of LOTOS and one of the main obstacles to widespread use.

We define two kinds of language extensions. Firstly, *virtual libraries* generalize the library mechanism of LOTOS in order to support a (virtual) infinite set of different types. Secondly, *syntax extensions* provide a more compact and more specific syntax for defining some commonly used data structures.

Both extensions are supported by a software processor that translate them into standard LOTOS. The translation program itself is composed of a *general-purpose text processor* that interprets external definitions of the translations to be performed. The benefit of this separation is twofold: the provided facilities can be extended or modified more easily, and different translations may be used for different underlying tools, thereby hiding their specific restrictions from the specifier. A new language, inspired from a syntax extension formalism used in Scheme, has been designed for defining the translations.

We then survey usual data type evaluation techniques and their respective limitations, and propose a *methodology for executable data specifications*, that meets usual restrictions and therefore ensures computability in most environments. The refined model associated with this methodology is close to functional programming.

The last contribution, of a more prospective nature, builds on the preceding results to propose a *new core data language* for LOTOS, of functional nature, which is a direct transposition of the refined model for executable data types. We discuss the relations with standard ACT ONE. We also present syntax enrichments for data values (e.g. list notations), as purely syntactical facilities independent from any particular type.

The achievements of this thesis are more in integrating, analyzing and exploiting existing knowledge than in particular theoretical contributions. Its main contributions are in the generic text-translation algorithm, its associated definition language and its usage to support the proposed LOTOS facilities, in the extraction of practical data type specification solutions from known theoretical properties of data type evaluation techniques and in the proposed new data language.