



Doctoral Thesis

Compiling the behaviour part of LOTOS

Author(s):

Dubuis, Eric Emile

Publication Date:

1993

Permanent Link:

<https://doi.org/10.3929/ethz-a-000887858> →

Rights / License:

[In Copyright - Non-Commercial Use Permitted](#) →

This page was generated automatically upon download from the [ETH Zurich Research Collection](#). For more information please consult the [Terms of use](#).

Diss. ETH No. 10027

Compiling the Behaviour Part of LOTOS

A dissertation submitted to the

SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH

for the degree of

Doctor of Technical Sciences

presented by

Eric Emile Dubuis
Dipl. El. Ing. ETH
born February 5th, 1956
citizen of Rossinière (VD)

accepted on the recommendation of

Prof. Dr. A. Kündig,	examiner
Prof. Dr. C. Petitpierre,	co-examiner
Dr. G. Karjoth,	co-examiner

1993

Abstract

LOTOS is a standardised formal description language to describe the behaviour of (information) systems. It has well-defined operational semantics which uniquely map a correct LOTOS specification onto a labeled transition system.

The formal framework of LOTOS allows the step-wise refinement of an initial abstract specification into an implementation-near form. Given an implementation-near form, then there is the problem of implementing an efficient system.

This thesis defines a novel execution model which we call *system of automata and ports* (SAP). It consists of automata and of ports. Automata represent the entities of a LOTOS description that have to be executed in parallel. Ports are resources whose purpose is to establish n -way synchronisations. The execution model is the starting point for the derivation of a concrete realisation.

The language LOTOS consists of two parts. One part is to define data types and operations. The other part is to define the dynamic behaviour of a system by using so-called behaviour expressions. We do not deal with the original data part of LOTOS in this work. Instead, we replace it by language constructs of the programming language C. Hence, we obtain an auxiliary language which we call C-LOTOS. C-LOTOS contains the most important behaviour operators of LOTOS except the disable operator, the general choice operators, and the general parallel operator. Also selection predicates in action-prefix expressions are not permitted.

We define a formal mapping for the translation of C-LOTOS into the execution model SAP by set of base cases and inference rules over the syntactical structure of C-LOTOS. Given a C-LOTOS specification, the repeated application of the base cases and the inference rules computes the execution model SAP.

To demonstrate the validity of the proposed approach, a translator and a run-time system are realised. The translator parses a C-LOTOS specification and builds an internal data structure representing SAP. A code generator then generates C code which in turn is compiled and linked with a suitably tailored run-time system. The product of this final step is an executable program of a C-LOTOS specification.

Zusammenfassung

LOTOS ist eine standardisierte formale Spezifikationsprache, um das Verhalten von (Informations-) Systemen zu beschreiben. Sie besitzt eine definierte, operationale Semantik. Diese ordnet zu jeder korrekten LOTOS-Spezifikation ein *labeled transition system* zu.

Der formale Rahmen von LOTOS ermöglicht, abstrakte Spezifikationen eines Systems schrittweise in eine implementationsnahe Form zu verfeinern. Liegt eine implementationsnahe Form vor, so stellt sich das Problem, ein effizientes System zu realisieren.

Vorliegende Arbeit definiert ein Ausführungsmodell, das ich *system of automata and ports* (SAP) nenne. Es besteht aus Automaten und Ports. Automaten verkörpern die Elemente einer LOTOS-Beschreibung, die parallel ausgeführt werden. Ports sind Ressourcen, mit denen *n*-Weg-Synchronisationen realisiert werden. Das Ausführungsmodell bildet den Ausgangspunkt zur Herleitung konkreter Realisationen.

LOTOS ist eine hybride Sprache. Datentypen und zugehörige Operationen werden mittels Signaturen und algebraischen Gleichungen definiert. Das dynamische Verhalten wird mit sogenannten Verhaltensausrücken beschrieben. Der Datenteil von LOTOS wird in der vorliegenden Arbeit ausgeklammert. An dessen Stelle führe ich Elemente der Programmiersprache C ein und erhalte so die Hilfsprache C-LOTOS. C-LOTOS umfasst die wichtigsten Verhaltensoperatoren von LOTOS. Nicht enthalten in C-LOTOS sind der *disable*-Operator, die *general choice*-Operatoren und der *general parallel*-Operator von LOTOS. Auch Selektionsprädikate in *action-prefix expressions* sind nicht zugelassen.

Ich definiere eine formale Abbildung, um C-LOTOS in das Ausführungsmodell SAP zu übersetzen. Die Abbildung ist mittels Basis- und Induktionsschritten über die syntaktische Struktur von C-LOTOS definiert. Ein wiederholtes Anwenden dieser Übersetzungsschritte übersetzt eine C-LOTOS-Spezifikation in das Ausführungsmodell SAP.

Die Richtigkeit des vorgeschlagenen Lösungsansatzes wird anhand eines Übersetzers und eines Laufzeitsystems demonstriert. Der Übersetzer zerlegt eine C-LOTOS-Spezifikation und bildet eine interne Datenstruktur, welche SAP entspricht. Ein Code-Generator generiert anschliessend C-Code. Der C-Code wird kompiliert und gelinkt. Das Resultat ist ein ausführbares Programm einer C-LOTOS-Spezifikation.