

A Formal Semantics of Data Flow Diagrams

Peter Gorm Larsen,¹ Nico Plat² and Hans Toetenel²

¹ IFAD, Odense, Denmark.

² Delft University of Technology, Faculty of Technical Mathematics and Informatics, Delft, The Netherlands. Nico Plat is currently with CAP Volmac, Utrecht, The Netherlands.

Keywords: Data flow diagrams; VDM; Formal semantics

Abstract. This paper presents a formal semantics of data flow diagrams as used in Structured Analysis, based on an abstract model for data flow transformations. The semantics consists of a collection of VDM functions, transforming an abstract syntax representation of a data flow diagram into an abstract syntax representation of a VDM specification. Since this transformation is executable, it becomes possible to provide a software analyst/designer with two ‘views’ of the system being modeled: a graphical view in terms of a data flow diagram, and a textual view in terms of a VDM specification. In this paper emphasis is on the motivation for the choices made in the transformation. The main aspects of the transformation itself are described using annotated VDM functions with some examples.

1. Introduction

The introduction of formal methods in industrial organizations may become easier if these methods can be used alongside the more widely used conventional techniques for software development, such as ‘structured methods’. Structured methods are methods for software analysis and design, based on the use of heuristics for making analysis and design decisions. They provide a relatively well-defined path, often in a cookbook-like fashion (hence the term ‘structured’ methods), starting from the analysis of software requirements and ending at system coding. The design notations used are usually graphical and have no formal basis. In that sense structured and formal methods can be regarded as complementary. It is often suggested that the informal graphical notations as provided by structured methods are intuitively appealing to software analysts/designers. Therefore, a combined structured/formal method may not only increase the understanding of the use of formal methods in the software process, but also may increase the acceptability of formal methods to these people.

Our work in this area so far has concentrated on combining *Structured Analysis (SA)* [You75, DeM79,

Correspondence and offprint requests to: Peter Gorm Larsen, IFAD, Forskerparken 10, DK-5230 Odense M, Denmark. E-mail: peter@ifad.dk