A Formal Approach to Hypertext using Post-Prototype Formal Specification

Danny B. Lange

Brüel & Kjær Industri A/S†
DK - 2850 Nærum, Denmark
Tel: +45 42 80 05 00
email: danny.lange@bk.dk

Department of Computer Science‡
Technical University of Denmark
DK - 2800 Lyngby, Denmark

Abstract

In this paper a formal specification of an object-oriented model of hypertext is presented. It is shown that formal specification is beneficial even in applications driven by user interface and implementation considerations. The formal specification is based on a prototype implementation of hypertext. The purpose was to gain insight and clarify hypertext systems architecture. Traditionally a prototype should be based on a set of formal specifications, or even better, the formal specification should be executable. It is argued that one can benefit from using formal specification actually after having made a prototype using less rigorous and more ad hoc techniques. Our actual experience with post-prototype formal specification supports that this approach gives a more precisely description of the domain and capture the nature of our experiences gained by operating the prototype.

1 Introduction

Hypertext is becoming a well-known technique for information representation and management. In the last few years an increasing number of papers on hypertext and its application has been published. Only a very small part of this work have been concerned

---

* A version of this paper emphasizing standardization of hypertext with different technical details, but inevitably overlapping in the datamodel facet with the present paper, has been presented at NIST Hypertext Standardization Workshop in Gaithersburg, Maryland USA (January 1990), and published in the workshop proceedings.
†Author’s Present Address
‡A part of the work has taken place at the Technical University of Denmark
with the formal treatment of hypertext. Hypertext has been popularized mainly through HyperCard\(^1\) and Hyperties [Michel 1989] [Shneiderman 1987], that may be purchased for use on personal computers. Different research projects shows that hypertext has many potentially applications which are just beginning to be explored: textbooks, dictionaries, encyclopedias and software engineering [Conklin 1987]. Basically, hypertext is an approach to information management in which data is stored in a network of nodes connected by links. The nodes are fragments of information, such as text and graphics. The links reflect relations between these information fragments. An user can browse a hypertext by traversing the network and view the information fragments as they are visited.

There is clearly a need for a more formal approach to hypertext since one can claim that hypertext is driven by user interface and implementation considerations. Attempts to present more formal models of hypertext is now beginning to appear [Garg 1988] [Delisle & Schwartz 1987] [Stotts & Furuta 1989]. This paper presents a formal specification of the hypertext system, using the Vienna Development Method (VDM) [Jones 1986] [Bjørner & Jones 1982].

Formally specification is becoming of increasing interest to industry. Industry has been slow to adopt formal methods. It is to commonly to meet statements like: “formal methods are only for use in critical software”, “my people haven’t been taught formal methods at university”, “we don’t have tools supporting formal development”, etc. In contrast to what most people think of when they talk about formal specification, which is the issues on formal refinement, proof obligations and refinement, this paper presents the use of formal specifications to gain insight into the system and help to clarify the system architecture.

This paper will attempt to show: (1) that formal specification is beneficial even in applications driven by user interface and implementation considerations, and (2) that formal specification can be used post-prototype. Using formal specification on well-known domains (that might be post-prototype) clearly gives a more precise description of the domain and capture the nature of the experiences. This should yield more valuable knowledge and insight into the problem domain. Hence formal specification should work as a mental tool or as an “intellectual testbed” for the project members. Furthermore it eases the communication between project members. This approach to formal specification is supported by the works of Delisle and Garlan on formally specifying electronic instruments [Delisle & Garlan 1989] and the studies of post-prototype formal specification in the Arcadia project [Terwilliger et al. 1989].

At Brüel & Kjær\(^2\) we have developed a prototype of a hypertext system. The prototype was developed on a SUN\(^3\) workstation\(^3\) using a system called ART\(^4\). The prototype was written partly in a rule-based language and COMMON LISP [Steele 1984] using a window based user interface, see figure 1. The prototype has fulfilled several aims. First it has given the developers a feeling of what hypertext is all about, by operating the prototype.

---

\(^1\)HyperCard is a registered trademark of Apple Computer, Inc.

\(^2\)Brüel & Kjær Industri is a company that designs and manufactures high-precision electronic measuring instruments.

\(^3\)Sun Workstation is a registered trademark of Sun Microsystems, Inc.

\(^4\)ART (Automated Reasoning Tool) is a registered trademark of Inference Corporation.