THESE de DOCTORAT de L'UNIVERSITE PARIS 6

Spécialité

INFORMATIQUE

présentée

par Mr **Jean-Francois BRETTE**

pour obtenir le grade de **DOCTEUR DE L'UNIVERSITE PARIS 6**

Sujet de la thèse :

PASCAL/V :

Un environnement pour l'apprentissage de la programmation par découverte guidée.

Réification et interprétation contextuelle du rôle des variables.

soutenue le 27 mai 1994
devant le jury composé de :

Mr Patrick BARRIL	
Mr Stefano CERRI	**Rapporteur**
Mr Raymond DURAND	
Mr Pascal ESTRAILLIER	
Mme Françoise MADAULE	**Directeur**
Mr Jean-François PERROT	**Président**
Mr Harald WERTZ	**Rapporteur**

Résumé

Cette thèse présente la conception et la réalisation de PASCAL/V, un environnement de programmation permettant d'articuler enseignement et apprentissage et centré sur le rôle des variables. Il comprend un compilateur pédagogique, un outil de visualisation de trace au pas-à-pas, des inspecteurs spécialisés pour les types et une aide contextuelle pour les erreurs d'exécution. Un grand soin a été porté à la fidélité conceptuelle des outils interactifs de découverte. Ils sont souvent le résultat d'un compromis entre besoins didactiques et possibilités techniques des interfaces.

Nous considérons que le rôle d'une variable est une extension de son type et fait le lien entre structure de données et traitements. L'implémentation fait clairement la distinction entre les vérifications de compatibilité de types, effectuées à la compilation et les vérifications de contraintes portant sur les sous-types, effectuées à l'exécution. Ces dernières sont la base de l'aide contextuelle qui fournit des suggestions à l'élève afin de le guider dans son apprentissage. Ces suggestions constituent un élément de la représentation du domaine sous forme de règles PROLOG/V et permettent notamment à l'élève de comprendre le rôle de l'environnement d'exécution dans l'interprétation d'une erreur.

PASCAL/V est écrit en SMALLTALK/V sur MacIntosh et sa réalisation nous a amené à explorer en détail les principes de compilation et de débogage interactif en SMALLTALK/V afin de construire différentiellement un compilateur Pascal, à expérimenter les limites de l'architecture d'interface utilisateur MVC de Smalltalk dont nous faisons une critique accompagnée de propositions nouvelles et à mettre en avant l'importance des coopérations de classes dans la programmation objet, notamment pour l'organisation système et la compréhension de code.

Mots-clés : Environnement d'apprentissage intelligent, programmation objet, environnement de programmation, didactique de la programmation, guidage contextuel, Smalltalk, Pascal.

Abstract

This thesis presents the design and the implementation of PASCAL/V, a programming environment allowing the articulation between teaching and learning and centered on the role of variables. It includes a pedagogical compiler, a step-by-step tracing visualisation tool, specialized inspectors for types and a contextual help for dynamic diagnosis. Great care has been taken over the conceptual fidelity of these interactive discovery tools. They are the result of an inevitable compromise between didactic needs and the technical possibilities of interfaces.

We consider that the role of a variable is an extension of its type and links data structures with treatments. The implementation makes a clear distinction between type compatibility checking, carried out at compilation time and constraints checking concerning subtypes, carried out at running time. The latter is the basis for the contextual help which gives suggestions to the student in order to guide him in his learning process. These suggestions constitute a part of the representation of the domain in the form of PROLOG/V rules and allow the student to understand the role of the running environment in the interpretation of errors.

PASCAL/V is written in SMALLTALK/V on MacIntosh and its development led us to explore in depth the principles of compilation and interactive debugging in SMALLTALK/V in order to build differentially a Pascal compiler ; to test the limits of Smalltalk's MVC user interface of which we offer our criticisms together with new propositions ; and to highlight the significance of class cooperations in object oriented programming, notably for system organization and code understanding.

Keywords : Intelligent learning environment, object oriented programming, programming environment, didactics of programming, contextual coaching, Smalltalk, Pascal.