Hartmut Ehrig
Gregor Engels
Francesco Parisi-Presicce
Grzegorz Rozenberg (Eds.)

LNCS 3256

Graph Transformations

Second International Conference, ICGT 2004
Rome, Italy, September/October 2004
Proceedings

Springer

Hartmut Ehrig Gregor Engels
Francesco Parisi-Presicce
Grzegorz Rozenberg (Eds.)

Graph Transformations

Second International Conference, ICGT 2004
Rome, Italy, September 28 – October 2, 2004
Proceedings

Springer

Volume Editors

Hartmut Ehrig
Technical University Berlin
Department for Software Technology and Theoretical Informatics
Sekr. FR 6-1, Franklinstr. 28/29, 10587 Berlin, Germany
E-mail: ehrig@cs.tu-berlin.de

Gregor Engels
University of Paderborn
Faculty of Computer Science, Electrical Engineering, and Mathematics
Warburger Str. 100, 33098 Paderborn, Germany
E-mail: engels@upb.de

Francesco Parisi-Presicce
George Mason University
Department of Information and Software Engineering
4400 University Drive, Fairfax, VA 22030, USA
E-mail: fparisip@gmu.edu

Grzegorz Rozenberg
Leiden University
Leiden Institute of Advanced Computer Science (LIACS)
Niels Bohrweg 1, 2333 CA Leiden, The Netherlands
rozenber@liacs.nl

Library of Congress Control Number: 2004112162

CR Subject Classification (1998): E.1, G.2.2, D.2.4, F.1, F.2.2, F.3, F.4.2-3

ISSN 0302-9743
ISBN 3-540-23207-9 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik
Printed on acid-free paper SPIN: 11325826 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3256

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

Preface

ICGT 2004 was the 2nd International Conference on Graph Transformation, following the first one in Barcelona (2002), and a series of six international workshops on graph grammars with applications in computer science between 1978 and 1998. ICGT 2004 was held in Rome (Italy), Sept. 29–Oct. 1, 2004 under the auspices of the European Association for Theoretical Computer Science (EATCS), the European Association of Software Science and Technology (EASST), and the IFIP WG 1.3, Foundations of Systems Specification.

The scope of the conference concerned graphical structures of various kinds (like graphs, diagrams, visual sentences and others) that are useful when describing complex structures and systems in a direct and intuitive way. These structures are often augmented with formalisms that add to the static description a further dimension, allowing for the modelling of the evolution of systems via all kinds of transformations of such graphical structures. The field of graph transformation is concerned with the theory, applications, and implementation issues of such formalisms.

The theory is strongly related to areas such as graph theory and graph algorithms, formal language and parsing theory, the theory of concurrent and distributed systems, formal specification and verification, logic, and semantics. The application areas include all those fields of computer science, information processing, engineering, and the natural sciences where static and dynamic modelling using graphical structures and graph transformations, respectively, play important roles. In many of these areas tools based on graph transformation technology have been implemented and used.

The proceedings of ICGT 2004 consist of two parts. The first part comprises the contributions of the invited talks followed by the carefully reviewed and accepted 26 papers that were selected out of 58 submissions. The topics of the papers range over a wide spectrum, including graph theory and graph algorithms, theoretic and semantic aspects, modelling, applications in chemistry and biology, and tool issues. The second part contains two tutorial introductions to graph transformation and their relation to software and DNA computing, and short presentations of the satellite events of ICGT 2004.

We would like to thank the members of the program committee and the secondary reviewers for their enormous help in the selection process. We are also grateful to Reiko Heckel and Alexey Cherchago for their technical support in running the conference system and in editing the proceedings. Moreover, we would like to express our gratitude to the local organizers Paolo Bottoni (Chair), and Marta Simeoni who did a great job. Finally, we would like to acknowledge the always excellent cooperation with Springer, the publisher of the Lecture Notes in Computer Science.

July 2004

Gregor Engels, Hartmut Ehrig
Francesco Parisi-Presicce, Grzegorz Rozenberg

Program Committee

Michel Bauderon	Bordeaux (France)
Dorothea Blostein	Kingston (Ontario, Canada)
Andrea Corradini	Pisa (Italy)
Hartmut Ehrig	Berlin (Germany)
Gregor Engels (co-chair)	Paderborn (Germany)
Reiko Heckel	Paderborn (Germany)
Dirk Janssens	Antwerp (Belgium)
Hans-Jörg Kreowski	Bremen (Germany)
Barbara König	Stuttgart (Germany)
Bernd Meyer	Clayton (Victoria, Australia)
Ugo Montanari	Pisa (Italy)
Manfred Nagl	Aachen (Germany)
Fernando Orejas	Barcelona (Spain)
Francesco Parisi-Presicce (co-chair)	Rome (Italy) and Fairfax (Virginia, USA)
Mauro Pezzè	Milan (Italy)
John Pfaltz	Charlottesville (Virginia, USA)
Rinus Plasmeijer	Nijmegen (The Netherlands)
Detlef Plump	York (UK)
Leila Ribeiro	Porto Alegre (Brazil)
Grzegorz Rozenberg	Leiden (The Netherlands)
Andy Schürr	Darmstadt (Germany)
Gabriele Taentzer	Berlin (Germany)
Genny Tortora	Salerno (Italy)
Gabriel Valiente	Barcelona (Spain)

Secondary Referees

Alon Amsel	Olaf Chitil	Jan Hendrik Hausmann
Zena Ariola	Juan de Lara	Tobias Heindel
Thomas Baeck	Juergen Dingel	Dan Hirsch
Paolo Baldan	Carlotta Domeniconi	Berthold Hoffmann
Luciano Baresi	Claudia Ermel	Kathrin Hoffmann
Stefan Blom	Martin Erwig	Jon Howse
Achim Blumensath	Alexander Förster	Karsten Hölscher
Boris Böhlen	Giorgio Ghelli	Johannes Jakob
Tommaso Bolognesi	Stefania Gnesi	Renate Klempien-Hinrichs
Paolo Bottoni	Martin Grosse-Rhode	Peter Knirsch
Antonio Brogi	Roberto Grossi	Maciej Koutny
Roberto Bruni	Szilvia Gyapay	Vitali Kozioura
Alexey Cherchago	Annegret Habel	Sabine Kuske

Georgios Lajios
Marc Lohmann
Kim Marriott
Antoni Mazurkiewicz
Antoine Meyer
Mohamed Mosbah
George Paun
Lucia Pomello
Ulricke Prange
Arend Rensink
Davide Rossi
Jörg Schneider
Stefan Schwoon
Pawel Sobocinski
Volker Sorge
James Stewart
Sebastian Thöne
Emilio Tuosto
Niels van Eetvelde
Pieter van Gorp
Dániel Varró
Bernhard Westfechtel
Hendrik Voigt
Dobieslaw Wroblenski
Takashi Yokomori

Sponsoring Institutions

The European Association for Theoretical Computer Science (EATCS), the European Association of Software Science and Technology (EASST), the IFIP Working Group 1.3, Foundations of Systems Specification, Università di Roma "La Sapienza", Provincia di Roma, Comune di Roma.

Lecture Notes in Computer Science

For information about Vols. 1–3130

please contact your bookseller or Springer

Vol. 3263: M. Weske, P. Liggesmeyer (Eds.), Object-Oriented and Internet-Based Technologies. XII, 239 pages. 2004.

Vol. 3260: I. Niemegeers, S.H. de Groot (Eds.), Personal Wireless Communications. XIV, 478 pages. 2004.

Vol. 3258: M. Wallace (Ed.), Principles and Practice of Constraint Programming – CP 2004. XVII, 822 pages. 2004.

Vol. 3256: H. Ehrig, G. Engels, F. Parisi-Presicce, G. Rozenberg (Eds.), Graph Transformations. XII, 451 pages. 2004.

Vol. 3255: A. Benczúr, J. Demetrovics, G. Gottlob (Eds.), Advances in Databases and Information Systems. XI, 423 pages. 2004.

Vol. 3254: E. Macii, V. Paliouras, O. Koufopavlou (Eds.), Integrated Circuit and System Design. XVI, 910 pages. 2004.

Vol. 3253: Y. Lakhnech, S. Yovine (Eds.), Formal Techniques in Timed, Real-Time, and Fault-Tolerant Systems. X, 397 pages. 2004.

Vol. 3250: L.-J. (LJ) Zhang, M. Jeckle (Eds.), Web Services. X, 300 pages. 2004.

Vol. 3249: B. Buchberger, J.A. Campbell (Eds.), Artificial Intelligence and Symbolic Computation. X, 285 pages. 2004. (Subseries LNAI).

Vol. 3246: A. Apostolico, M. Melucci (Eds.), String Processing and Information Retrieval. XIV, 316 pages. 2004.

Vol. 3242: X. Yao (Ed.), Parallel Problem Solving from Nature - PPSN VIII. XX, 1185 pages. 2004.

Vol. 3241: D. Kranzlmüller, P. Kacsuk, J.J. Dongarra (Eds.), Recent Advances in Parallel Virtual Machine and Message Passing Interface. XIII, 452 pages. 2004.

Vol. 3240: I. Jonassen, J. Kim (Eds.), Algorithms in Bioinformatics. IX, 476 pages. 2004. (Subseries LNBI).

Vol. 3239: G. Nicosia, V. Cutello, P.J. Bentley, J. Timmis (Eds.), Artificial Immune Systems. XII, 444 pages. 2004.

Vol. 3238: S. Biundo, T. Frühwirth, G. Palm (Eds.), KI 2004: Advances in Artificial Intelligence. XI, 467 pages. 2004. (Subseries LNAI).

Vol. 3232: R. Heery, L. Lyon (Eds.), Research and Advanced Technology for Digital Libraries. XV, 528 pages. 2004.

Vol. 3229: J.J. Alferes, J. Leite (Eds.), Logics in Artificial Intelligence. XIV, 744 pages. 2004. (Subseries LNAI).

Vol. 3224: E. Jonsson, A. Valdes, M. Almgren (Eds.), Recent Advances in Intrusion Detection. XII, 315 pages. 2004.

Vol. 3223: K. Slind, A. Bunker, G. Gopalakrishnan (Eds.), Theorem Proving in Higher Order Logics. VIII, 337 pages. 2004.

Vol. 3221: S. Albers, T. Radzik (Eds.), Algorithms – ESA 2004. XVIII, 836 pages. 2004.

Vol. 3220: J.C. Lester, R.M. Vicari, F. Paraguaçu (Eds.), Intelligent Tutoring Systems. XXI, 920 pages. 2004.

Vol. 3217: C. Barillot, D.R. Haynor, P. Hellier (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2004. XXXVIII, 1114 pages. 2004.

Vol. 3216: C. Barillot, D.R. Haynor, P. Hellier (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2004. XXXVIII, 930 pages. 2004.

Vol. 3210: J. Marcinkowski, A. Tarlecki (Eds.), Computer Science Logic. XI, 520 pages. 2004.

Vol. 3208: H.J. Ohlbach, S. Schaffert (Eds.), Principles and Practice of Semantic Web Reasoning. VII, 165 pages. 2004.

Vol. 3207: L.T. Yang, M. Guo, G.R. Gao, N.K. Jha (Eds.), Embedded and Ubiquitous Computing. XX, 1116 pages. 2004.

Vol. 3206: P. Sojka, I. Kopecek, K. Pala (Eds.), Text, Speech and Dialogue. XIII, 667 pages. 2004. (Subseries LNAI).

Vol. 3205: N. Davies, E. Mynatt, I. Siio (Eds.), UbiComp 2004: Ubiquitous Computing. XVI, 452 pages. 2004.

Vol. 3203: J. Becker, M. Platzner, S. Vernalde (Eds.), Field Programmable Logic and Application. XXX, 1198 pages. 2004.

Vol. 3202: J.-F. Boulicaut, F. Esposito, F. Giannotti, D. Pedreschi (Eds.), Knowledge Discovery in Databases: PKDD 2004. XIX, 560 pages. 2004. (Subseries LNAI).

Vol. 3201: J.-F. Boulicaut, F. Esposito, F. Giannotti, D. Pedreschi (Eds.), Machine Learning: ECML 2004. XVIII, 580 pages. 2004. (Subseries LNAI).

Vol. 3199: H. Schepers (Ed.), Software and Compilers for Embedded Systems. X, 259 pages. 2004.

Vol. 3198: G.-J. de Vreede, L.A. Guerrero, G. Marín Raventós (Eds.), Groupware: Design, Implementation and Use. XI, 378 pages. 2004.

Vol. 3194: R. Camacho, R. King, A. Srinivasan (Eds.), Inductive Logic Programming. XI, 361 pages. 2004. (Subseries LNAI).

Vol. 3193: P. Samarati, P. Ryan, D. Gollmann, R. Molva (Eds.), Computer Security – ESORICS 2004. X, 457 pages. 2004.

Vol. 3192: C. Bussler, D. Fensel (Eds.), Artificial Intelligence: Methodology, Systems, and Applications. XIII, 522 pages. 2004. (Subseries LNAI).

Vol. 3190: Y. Luo (Ed.), Cooperative Design, Visualization, and Engineering. IX, 248 pages. 2004.

Vol. 3189: P.-C. Yew, J. Xue (Eds.), Advances in Computer Systems Architecture. XVII, 598 pages. 2004.

Vol. 3186: Z. Bellahsène, T. Milo, M. Rys, D. Suciu, R. Unland (Eds.), Database and XML Technologies. X, 235 pages. 2004.

Vol. 3185: M. Bernardo, F. Corradini (Eds.), Formal Methods for the Design of Real-Time Systems. VII, 295 pages. 2004.

Vol. 3184: S. Katsikas, J. Lopez, G. Pernul (Eds.), Trust and Privacy in Digital Business. XI, 299 pages. 2004.

Vol. 3183: R. Traunmüller (Ed.), Electronic Government. XIX, 583 pages. 2004.

Vol. 3182: K. Bauknecht, M. Bichler, B. Pröll (Eds.), E-Commerce and Web Technologies. XI, 370 pages. 2004.

Vol. 3181: Y. Kambayashi, M. Mohania, W. Wöß (Eds.), Data Warehousing and Knowledge Discovery. XIV, 412 pages. 2004.

Vol. 3180: F. Galindo, M. Takizawa, R. Traunmüller (Eds.), Database and Expert Systems Applications. XXI, 972 pages. 2004.

Vol. 3179: F.J. Perales, B.A. Draper (Eds.), Articulated Motion and Deformable Objects. XI, 270 pages. 2004.

Vol. 3178: W. Jonker, M. Petkovic (Eds.), Secure Data Management. VIII, 219 pages. 2004.

Vol. 3177: Z.R. Yang, H. Yin, R. Everson (Eds.), Intelligent Data Engineering and Automated Learning – IDEAL 2004. XVIII, 852 pages. 2004.

Vol. 3176: O. Bousquet, U. von Luxburg, G. Rätsch (Eds.), Advanced Lectures on Machine Learning. IX, 241 pages. 2004. (Subseries LNAI).

Vol. 3175: C.E. Rasmussen, H.H. Bülthoff, B. Schölkopf, M.A. Giese (Eds.), Pattern Recognition. XVIII, 581 pages. 2004.

Vol. 3174: F. Yin, J. Wang, C. Guo (Eds.), Advances in Neural Networks - ISNN 2004. XXXV, 1021 pages. 2004.

Vol. 3173: F. Yin, J. Wang, C. Guo (Eds.), Advances in Neural Networks – ISNN 2004. XXXV, 1041 pages. 2004.

Vol. 3172: M. Dorigo, M. Birattari, C. Blum, L. M. Gambardella, F. Mondada, T. Stützle (Eds.), Ant Colony, Optimization and Swarm Intelligence. XII, 434 pages. 2004.

Vol. 3170: P. Gardner, N. Yoshida (Eds.), CONCUR 2004 - Concurrency Theory. XIII, 529 pages. 2004.

Vol. 3166: M. Rauterberg (Ed.), Entertainment Computing – ICEC 2004. XXIII, 617 pages. 2004.

Vol. 3163: S. Marinai, A. Dengel (Eds.), Document Analysis Systems VI. XI, 564 pages. 2004.

Vol. 3162: R. Downey, M. Fellows, F. Dehne (Eds.), Parameterized and Exact Computation. X, 293 pages. 2004.

Vol. 3160: S. Brewster, M. Dunlop (Eds.), Mobile Human-Computer Interaction – MobileHCI 2004. XVII, 541 pages. 2004.

Vol. 3159: U. Visser, Intelligent Information Integration for the Semantic Web. XIV, 150 pages. 2004. (Subseries LNAI).

Vol. 3158: I. Nikolaidis, M. Barbeau, E. Kranakis (Eds.), Ad-Hoc, Mobile, and Wireless Networks. IX, 344 pages. 2004.

Vol. 3157: C. Zhang, H. W. Guesgen, W.K. Yeap (Eds.), PRICAI 2004: Trends in Artificial Intelligence. XX, 1023 pages. 2004. (Subseries LNAI).

Vol. 3156: M. Joye, J.-J. Quisquater (Eds.), Cryptographic Hardware and Embedded Systems - CHES 2004. XIII, 455 pages. 2004.

Vol. 3155: P. Funk, P.A. González Calero (Eds.), Advances in Case-Based Reasoning. XIII, 822 pages. 2004. (Subseries LNAI).

Vol. 3154: R.L. Nord (Ed.), Software Product Lines. XIV, 334 pages. 2004.

Vol. 3153: J. Fiala, V. Koubek, J. Kratochvíl (Eds.), Mathematical Foundations of Computer Science 2004. XIV, 902 pages. 2004.

Vol. 3152: M. Franklin (Ed.), Advances in Cryptology – CRYPTO 2004. XI, 579 pages. 2004.

Vol. 3150: G.-Z. Yang, T. Jiang (Eds.), Medical Imaging and Augmented Reality. XII, 378 pages. 2004.

Vol. 3149: M. Danelutto, M. Vanneschi, D. Laforenza (Eds.), Euro-Par 2004 Parallel Processing. XXXIV, 1081 pages. 2004.

Vol. 3148: R. Giacobazzi (Ed.), Static Analysis. XI, 393 pages. 2004.

Vol. 3147: H. Ehrig, W. Damm, J. Desel, M. Große-Rhode, W. Reif, E. Schnieder, E. Westkämper (Eds.), Integration of Software Specification Techniques for Applications in Engineering. X, 628 pages. 2004.

Vol. 3146: P. Érdi, A. Esposito, M. Marinaro, S. Scarpetta (Eds.), Computational Neuroscience: Cortical Dynamics. XI, 161 pages. 2004.

Vol. 3144: M. Papatriantafilou, P. Hunel (Eds.), Principles of Distributed Systems. XI, 246 pages. 2004.

Vol. 3143: W. Liu, Y. Shi, Q. Li (Eds.), Advances in Web-Based Learning – ICWL 2004. XIV, 459 pages. 2004.

Vol. 3142: J. Diaz, J. Karhumäki, A. Lepistö, D. Sannella (Eds.), Automata, Languages and Programming. XIX, 1253 pages. 2004.

Vol. 3140: N. Koch, P. Fraternali, M. Wirsing (Eds.), Web Engineering. XXI, 623 pages. 2004.

Vol. 3139: F. Iida, R. Pfeifer, L. Steels, Y. Kuniyoshi (Eds.), Embodied Artificial Intelligence. IX, 331 pages. 2004. (Subseries LNAI).

Vol. 3138: A. Fred, T. Caelli, R.P.W. Duin, A. Campilho, D.d. Ridder (Eds.), Structural, Syntactic, and Statistical Pattern Recognition. XXII, 1168 pages. 2004.

Vol. 3137: P. De Bra, W. Nejdl (Eds.), Adaptive Hypermedia and Adaptive Web-Based Systems. XIV, 442 pages. 2004.

Vol. 3136: F. Meziane, E. Métais (Eds.), Natural Language Processing and Information Systems. XII, 436 pages. 2004.

Vol. 3134: C. Zannier, H. Erdogmus, L. Lindstrom (Eds.), Extreme Programming and Agile Methods - XP/Agile Universe 2004. XIV, 233 pages. 2004.

Vol. 3133: A.D. Pimentel, S. Vassiliadis (Eds.), Computer Systems: Architectures, Modeling, and Simulation. XIII, 562 pages. 2004.

Vol. 3132: B. Demoen, V. Lifschitz (Eds.), Logic Programming. XII, 480 pages. 2004.

Vol. 3131: V. Torra, Y. Narukawa (Eds.), Modeling Decisions for Artificial Intelligence. XI, 327 pages. 2004. (Subseries LNAI).

Table of Contents

Invited Papers

Integration Technology

Chemistry and Biology

Graph Transformation Concepts

DPO Theory for High-Level Structures

Analysis and Testing

Graph Theory and Algorithms

Application Conditions and Logic

Transformation of Special Structures

Object-Orientation

Tutorials and Workshops

Improving Flow in Software Development Through Graphical Representations*

Margaret-Anne D. Storey

University of Victoria, British Columbia, Canada
mstorey@uvic.ca

Abstract. Software development is a challenging and time intensive task that requires much tool support to enhance software comprehension and collaborative work in software engineering. Many of the popular tools used in industry offer simple, yet highly effective, graphical aids to enhance programming tasks. In particular, tree views are frequently used to present features in the software and to facilitate navigation. General graph layouts, popular in many academic tools, are seen less frequently in industrial software development tools. Interactive graphs can allow a developer to visualize and manipulate non-structural relationships and abstractions in the software. In this presentation, I explore how graphical techniques developed in academia can improve "flow" for programmers using industrial development tools. The theory of "flow and optimal experiences" is used to offer rich explanations for the existence of many typical software tool features and to illuminate areas for potential improvements from graphical tool support.

* An extended version of this abstract is published in the IEEE proceedings of VL/HCC'04 (IEEE Symposium on Visual Languages and Human-Centric Computing), Rome, Italy, September 26-29, 2004.

H. Ehrig et al. (Eds.): ICGT 2004, LNCS 3256, p. 1, 2004.

A Perspective on Graphs and Access Control Models

Ravi Sandhu

George Mason University and NSD Security
ISE Department, MS4A4
George Mason University
Fairfax, VA 22030, USA
sandhu@gmu.edu
http://www.list.gmu.edu

Abstract. There would seem to be a natural connection between graphs and information security. This is particularly so in the arena of access control and authorization. Research on applying graph theory to access control problems goes back almost three decades. Nevertheless it is yet to make its way into the mainstream of access control research and practice. Much of this prior research is based on first principles, although more recently there have been significant efforts to build upon existing graph theory results and approaches. This paper gives a perspective on some of the connections between graphs and their transformations and access control models, particularly with respect to the safety problem and dynamic role hierarchies.

1 Introduction

In concept there appears to be a strong potential for graphs and their transformations to be applied to information security problems. In practice, however, this potential largely remains to be realized. Applications of graph theory in the security domain go back almost three decades and there has been a steady trickle of papers exploring this potential. Nonetheless graph theory has yet to make its way into the mainstream of security research and practice. In part this may be due to the relative youth of the security discipline and the particular focus of the research community in the early years. Because of the versatility of graph representations and graph theory techniques perhaps it is only a matter of time before a strong and compelling connection is found.

Information security is a broad field and offers multiple avenues for application of graph theory. To pick just two examples, in recent years we have seen application of graph theory in penetration testing and vulnerability analysis [2, 7, 17, 20, 29] and in authentication metrics [21]. It is beyond the scope of this paper to consider the vast landscape of information security. Rather we will focus on the specific area of access control and authorization.

We begin with a brief review of access control and access control models, and then identify two specific problems of access control where graph theory has been

H. Ehrig et al. (Eds.): ICGT 2004, LNCS 3256, pp. 2–12, 2004.

employed in the past. These are the so-called safety problem and the problem of dynamic hierarchies. The rest of the paper explores past work in these two problem areas in some detail and concludes with a brief discussion of possible future research.

Access Control

Access control is concerned with the question of who can do what in a computer system. Clearly the same object (such as a file) may be accessible by different users in different ways. Some users may be able to read and write the file, others to just read it and still others who have no access to the file. Strictly speaking users do not manipulate files directly but rather do so via programs (such as a text editor or word processor). A program executing on behalf of a user is called a subject, so access control is concerned with enforcing authorized access of subjects to objects. This basic idea was introduced by Lampson in a classic paper [14] and continues to be the central abstraction of access control. Authorization in Lampson's access matrix model is determined by access rights (such as r for read and w for write) in the cells of an access matrix. An example of an access matrix is shown in figure 1. Here subject U can read and write file F but only read file G. Subject V can read and write file G but has no access to file F. A review of the essential concepts of access control is available in [25].

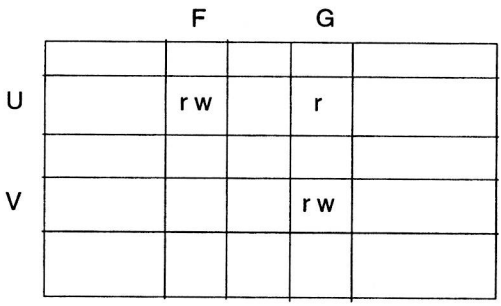

Fig. 1. Example of an Access Matrix.

The access matrix of figure 1 can be easily depicted as a directed graph with labelled edges as shown in figure 2. Thereby the intuitive feeling that there is a strong connection between graphs and access control. For convenience, we will henceforth talk of the access matrix and access graph as equivalent notions.

Access Control Models

A static access graph is not very interesting. Real computer systems are highly dynamic in that the access rights of subjects to objects change over time and new subjects and objects (and thereby new rights) are created and existing ones

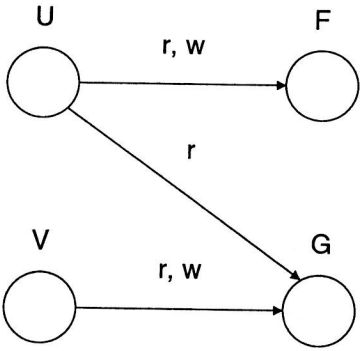

Fig. 2. Example of an Access Graph.

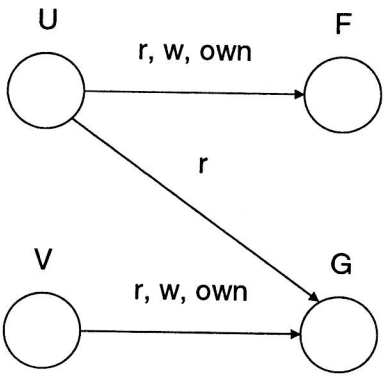

Fig. 3. Owner-Based Discretionary Access Control.

deleted. In terms of the access matrix this means that not only can contents of existing cells be changed but new rows and columns can be created and existing ones destroyed. In terms of the access graph, in addition to edge adding and deleting operations new nodes can be created and existing ones deleted.

An access control model specifies the operations by which the access graph can be changed. These operations are typically authorized by existing rights in the access graph itself. A common example of this is the "own" right shown in figure 3. The owner of a file has the own right for it and can add and delete rights for that file at the owner's free discretion. Thus subjects U and V control the rights of all subjects to files F and G respectively, i.e., U and V control the addition and deletion of edges labelled r or w terminating in F and G respectively.

The policy of owner-based discretionary access control is certainly reasonable but researchers quickly realized that there are many other policies of practical interest. For example, can the "own" right itself be granted? Some systems do not allow this. The creator of a file becomes its owner and remains its owner thereafter. Other systems allow ownership to be propagated from one subject to