Henning Christiansen
Mohand-Said Hacid
Troels Andreasen
Henrik Legind Larsen (Eds.)

Flexible Query Answering Systems

LNAI 3055

6th International Conference, FQAS 2004
Lyon, France, June 2004
Proceedings

Springer

Henning Christiansen Mohand-Said Hacid
Troels Andreasen Henrik Legind Larsen (Eds.)

Flexible Query Answering Systems

6th International Conference, FQAS 2004
Lyon, France, June 24-26, 2004
Proceedings

Springer

Volume Editors

Henning Christiansen
Roskilde University, Computer Science Dept.
P.O. Box 260, 4000 Roskilde, Denmark
E-mail: henning@ruc.dk

Mohand-Said Hacid
Université Lyon 1, LIRIS - UFR d'Informatique
43 boulevard du 11 novembre 1918
69622 Villeurbanne cedex, France
E-mail: mshacid@liris.univ-lyon1.fr

Troels Andreasen
Roskilde University, Department of Computer Science
P.O. Box 260, 4000 Roskilde, Denmark
E-mail: troels@ruc.dk

Henrik Legind Larsen
Aalborg University Esbjerg, Department of Computer Science and Engineering
Niels Bohrs Vej 8, 6700 Esbjerg, Denmark
E-mail: legind@cs.aue.auc.dk

Library of Congress Control Number: 2004106992

CR Subject Classification (1998): I.2, H.3, H.2, H.4, H.5

ISSN 0302-9743
ISBN 3-540-22160-3 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik
Printed on acid-free paper SPIN: 11010753 06/3142 5 4 3 2 1 0

Lecture Notes in Artificial Intelligence 3055

Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Springer
Berlin
Heidelberg
New York
Hong Kong
London
Milan
Paris
Tokyo

Lecture Notes in Artificial Intelligence (LNAI)

Vol. 3070: L. Rutkowski, J. Siekmann, R. Tadeusiewicz, L.A. Zadeh (Eds.), Artificial Intelligence and Soft Computing - ICAISC 2004. XXV, 1208 pages. 2004.

Vol. 3068: E. André, L. Dybkj\ae r, W. Minker, P. Heisterkamp (Eds.), Affective Dialogue Systems. XII, 324 pages. 2004.

Vol. 3067: M. Dastani, J. Dix, A. El Fallah-Seghrouchni (Eds.), Programming Multi-Agent Systems. X, 221 pages. 2004.

Vol. 3066: S. Tsumoto, R. S lowiński, J. Komorowski, J.W. Grzymala-Busse (Eds.), Rough Sets and Current Trends in Computing. XX, 853 pages. 2004.

Vol. 3065: A. Lomuscio, D. Nute (Eds.), Deontic Logic in Computer Science. X, 275 pages. 2004.

Vol. 3060: A.Y. Tawfik, S.D. Goodwin (Eds.), Advances in Artificial Intelligence. XIII, 582 pages. 2004.

Vol. 3056: H. Dai, R. Srikant, C. Zhang (Eds.), Advances in Knowledge Discovery and Data Mining. XIX, 713 pages. 2004.

Vol. 3055: H. Christiansen, M.-S. Hacid, T. Andreasen, H.L. Larsen (Eds.), Flexible Query Answering Systems. X, 500 pages. 2004.

Vol. 3040: R. Conejo, M. Urretavizcaya, J.-L. Pérez-de-la-Cruz (Eds.), Current Topics in Artificial Intelligence. XIV, 689 pages. 2004.

Vol. 3035: M.A. Wimmer (Ed.), Knowledge Management in Electronic Government. XII, 326 pages. 2004.

Vol. 3034: J. Favela, E. Menasalvas, E. Chávez (Eds.), Advances in Web Intelligence. XIII, 227 pages. 2004.

Vol. 3030: P. Giorgini, B. Henderson-Sellers, M. Winikoff (Eds.), Agent-Oriented Information Systems. XIV, 207 pages. 2004.

Vol. 3029: B. Orchard, C. Yang, M. Ali (Eds.), Innovations in Applied Artificial Intelligence. XXI, 1272 pages. 2004.

Vol. 3025: G.A. Vouros, T. Panayiotopoulos (Eds.), Methods and Applications of Artificial Intelligence. XV, 546 pages. 2004.

Vol. 3012: K. Kurumatani, S.-H. Chen, A. Ohuchi (Eds.), Multi-Agnets for Mass User Support. X, 217 pages. 2004.

Vol. 3010: K.R. Apt, F. Fages, F. Rossi, P. Szeredi, J. Váncza (Eds.), Recent Advances in Constraints. VIII, 285 pages. 2004.

Vol. 2990: J. Leite, A. Omicini, L. Sterling, P. Torroni (Eds.), Declarative Agent Languages and Technologies. XII, 281 pages. 2004.

Vol. 2980: A. Blackwell, K. Marriott, A. Shimojima (Eds.), Diagrammatic Representation and Inference. XV, 448 pages. 2004.

Vol. 2977: G. Di Marzo Serugendo, A. Karageorgos, O.F. Rana, F. Zambonelli (Eds.), Engineering Self-Organising Systems. X, 299 pages. 2004.

Vol. 2972: R. Monroy, G. Arroyo-Figueroa, L.E. Sucar, H. Sossa (Eds.), MICAI 2004: Advances in Artificial Intelligence. XVII, 923 pages. 2004.

Vol. 2961: P. Eklund (Ed.), Concept Lattices. IX, 411 pages. 2004.

Vol. 2953: K. Konrad, Model Generation for Natural Language Interpretation and Analysis. XIII, 166 pages. 2004.

Vol. 2934: G. Lindemann, D. Moldt, M. Paolucci (Eds.), Regulated Agent-Based Social Systems. X, 301 pages. 2004.

Vol. 2930: F. Winkler (Ed.), Automated Deduction in Geometry. VII, 231 pages. 2004.

Vol. 2926: L. van Elst, V. Dignum, A. Abecker (Eds.), Agent-Mediated Knowledge Management. XI, 428 pages. 2004.

Vol. 2923: V. Lifschitz, I. Niemelä (Eds.), Logic Programming and Nonmonotonic Reasoning. IX, 365 pages. 2004.

Vol. 2915: A. Camurri, G. Volpe (Eds.), Gesture-Based Communication in Human-Computer Interaction. XIII, 558 pages. 2004.

Vol. 2913: T.M. Pinkston, V.K. Prasanna (Eds.), High Performance Computing - HiPC 2003. XX, 512 pages. 2003.

Vol. 2903: T.D. Gedeon, L.C.C. Fung (Eds.), AI 2003: Advances in Artificial Intelligence. XVI, 1075 pages. 2003.

Vol. 2902: F.M. Pires, S.P. Abreu (Eds.), Progress in Artificial Intelligence. XV, 504 pages. 2003.

Vol. 2892: F. Dau, The Logic System of Concept Graphs with Negation. XI, 213 pages. 2003.

Vol. 2891: J. Lee, M. Barley (Eds.), Intelligent Agents and Multi-Agent Systems. X, 215 pages. 2003.

Vol. 2882: D. Veit, Matchmaking in Electronic Markets. XV, 180 pages. 2003.

Vol. 2871: N. Zhong, Z.W. Raś, S. Tsumoto, E. Suzuki (Eds.), Foundations of Intelligent Systems. XV, 697 pages. 2003.

Vol. 2854: J. Hoffmann, Utilizing Problem Structure in Planing. XIII, 251 pages. 2003.

Vol. 2843: G. Grieser, Y. Tanaka, A. Yamamoto (Eds.), Discovery Science. XII, 504 pages. 2003.

Vol. 2842: R. Gavaldá, K.P. Jantke, E. Takimoto (Eds.), Algorithmic Learning Theory. XI, 313 pages. 2003.

Vol. 2838: N. Lavrač, D. Gamberger, L. Todorovski, H. Blockeel (Eds.), Knowledge Discovery in Databases: PKDD 2003. XVI, 508 pages. 2003.

Vol. 2837: N. Lavrač, D. Gamberger, L. Todorovski, H. Blockeel (Eds.), Machine Learning: ECML 2003. XVI, 504 pages. 2003.

Vol. 2835: T. Horváth, A. Yamamoto (Eds.), Inductive Logic Programming. X, 401 pages. 2003.

Vol. 2821: A. Günter, R. Kruse, B. Neumann (Eds.), KI 2003: Advances in Artificial Intelligence. XII, 662 pages. 2003.

Vol. 2807: V. Matoušek, P. Mautner (Eds.), Text, Speech and Dialogue. XIII, 426 pages. 2003.

Vol. 2801: W. Banzhaf, J. Ziegler, T. Christaller, P. Dittrich, J.T. Kim (Eds.), Advances in Artificial Life. XVI, 905 pages. 2003.

Vol. 2797: O.R. Zaïane, S.J. Simoff, C. Djeraba (Eds.), Mining Multimedia and Complex Data. XII, 281 pages. 2003.

Vol. 2792: T. Rist, R.S. Aylett, D. Ballin, J. Rickel (Eds.), Intelligent Virtual Agents. XV, 364 pages. 2003.

Vol. 2782: M. Klusch, A. Omicini, S. Ossowski, H. Laamanen (Eds.), Cooperative Information Agents VII. XI, 345 pages. 2003.

Vol. 2780: M. Dojat, E. Keravnou, P. Barahona (Eds.), Artificial Intelligence in Medicine. XIII, 388 pages. 2003.

Vol. 2777: B. Schölkopf, M.K. Warmuth (Eds.), Learning Theory and Kernel Machines. XIV, 746 pages. 2003.

Vol. 2752: G.A. Kaminka, P.U. Lima, R. Rojas (Eds.), RoboCup 2002: Robot Soccer World Cup VI. XVI, 498 pages. 2003.

Vol. 2741: F. Baader (Ed.), Automated Deduction – CADE-19. XII, 503 pages. 2003.

Vol. 2705: S. Renals, G. Grefenstette (Eds.), Text- and Speech-Triggered Information Access. VII, 197 pages. 2003.

Vol. 2703: O.R. Zaïane, J. Srivastava, M. Spiliopoulou, B. Masand (Eds.), WEBKDD 2002 - MiningWeb Data for Discovering Usage Patterns and Profiles. IX, 181 pages. 2003.

Vol. 2700: M.T. Pazienza (Ed.), Extraction in the Web Era. XIII, 163 pages. 2003.

Vol. 2699: M.G. Hinchey, J.L. Rash, W.F. Truszkowski, C.A. Rouff, D.F. Gordon-Spears (Eds.), Formal Approaches to Agent-Based Systems. IX, 297 pages. 2002.

Vol. 2691: V. Mařík, J.P. Müller, M. Pechoucek (Eds.), Multi-Agent Systems and Applications III. XIV, 660 pages. 2003.

Vol. 2684: M.V. Butz, O. Sigaud, P. Gérard (Eds.), Anticipatory Behavior in Adaptive Learning Systems. X, 303 pages. 2003.

Vol. 2671: Y. Xiang, B. Chaib-draa (Eds.), Advances in Artificial Intelligence. XIV, 642 pages. 2003.

Vol. 2663: E. Menasalvas, J. Segovia, P.S. Szczepaniak (Eds.), Advances in Web Intelligence. XII, 350 pages. 2003.

Vol. 2661: P.L. Lanzi, W. Stolzmann, S.W. Wilson (Eds.), Learning Classifier Systems. VII, 231 pages. 2003.

Vol. 2654: U. Schmid, Inductive Synthesis of Functional Programs. XXII, 398 pages. 2003.

Vol. 2650: M.-P. Huget (Ed.), Communications in Multiagent Systems. VIII, 323 pages. 2003.

Vol. 2645: M.A. Wimmer (Ed.), Knowledge Management in Electronic Government. XI, 320 pages. 2003.

Vol. 2639: G. Wang, Q. Liu, Y. Yao, A. Skowron (Eds.), Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing. XVII, 741 pages. 2003.

Vol. 2637: K.-Y. Whang, J. Jeon, K. Shim, J. Srivastava, Advances in Knowledge Discovery and Data Mining. XVIII, 610 pages. 2003.

Vol. 2636: E. Alonso, D. Kudenko, D. Kazakov (Eds.), Adaptive Agents and Multi-Agent Systems. XIV, 323 pages. 2003.

Vol. 2627: B. O'Sullivan (Ed.), Recent Advances in Constraints. X, 201 pages. 2003.

Vol. 2600: S. Mendelson, A.J. Smola (Eds.), Advanced Lectures on Machine Learning. IX, 259 pages. 2003.

Vol. 2592: R. Kowalczyk, J.P. Müller, H. Tianfield, R. Unland (Eds.), Agent Technologies, Infrastructures, Tools, and Applications for E-Services. XVII, 371 pages. 2003.

Vol. 2586: M. Klusch, S. Bergamaschi, P. Edwards, P. Petta (Eds.), Intelligent Information Agents. VI, 275 pages. 2003.

Vol. 2583: S. Matwin, C. Sammut (Eds.), Inductive Logic Programming. X, 351 pages. 2003.

Vol. 2581: J.S. Sichman, F. Bousquet, P. Davidsson (Eds.), Multi-Agent-Based Simulation. X, 195 pages. 2003.

Vol. 2577: P. Petta, R. Tolksdorf, F. Zambonelli (Eds.), Engineering Societies in the Agents World III. X, 285 pages. 2003.

Vol. 2569: D. Karagiannis, U. Reimer (Eds.), Practical Aspects of Knowledge Management. XIII, 648 pages. 2002.

Vol. 2560: S. Goronzy, Robust Adaptation to Non-Native Accents in Automatic Speech Recognition. XI, 144 pages. 2002.

Vol. 2557: B. McKay, J. Slaney (Eds.), AI 2002: Advances in Artificial Intelligence. XV, 730 pages. 2002.

Vol. 2554: M. Beetz, Plan-Based Control of Robotic Agents. XI, 191 pages. 2002.

Vol. 2543: O. Bartenstein, U. Geske, M. Hannebauer, O. Yoshie (Eds.), Web Knowledge Management and Decision Support. X, 307 pages. 2003.

Vol. 2541: T. Barkowsky, Mental Representation and Processing of Geographic Knowledge. X, 174 pages. 2002.

Vol. 2533: N. Cesa-Bianchi, M. Numao, R. Reischuk (Eds.), Algorithmic Learning Theory. XI, 415 pages. 2002.

Vol. 2531: J. Padget, O. Shehory, D. Parkes, N.M. Sadeh, W.E. Walsh (Eds.), Agent-Mediated Electronic Commerce IV. Designing Mechanisms and Systems. XVII, 341 pages. 2002.

Vol. 2527: F.J. Garijo, J.-C. Riquelme, M. Toro (Eds.), Advances in Artificial Intelligence - IBERAMIA 2002. XVIII, 955 pages. 2002.

Vol. 2522: T. Andreasen, A. Motro, H. Christiansen, H.L. Larsen (Eds.), Flexible Query Answering Systems. X, 383 pages. 2002.

Vol. 2514: M. Baaz, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning. XIII, 465 pages. 2002.

Vol. 2507: G. Bittencourt, G.L. Ramalho (Eds.), Advances in Artificial Intelligence. XIII, 417 pages. 2002.

Preface

This volume constitutes the proceedings of the Sixth International Conference on Flexible Query Answering Systems, FQAS 2004, held in Lyon, France, on June 24–26, 2004. FQAS is the premier conference for researchers and practitioners concerned with the vital task of providing easy, flexible, and intuitive access to information for every type of need. This multidisciplinary conference draws on several research areas, including databases, information retrieval, knowledge representation, soft computing, multimedia, and human-computer interaction. With FQAS 2004, the FQAS conference series celebrated its tenth anniversary as it has been held every two years since 1994. The overall theme of the FQAS conferences is innovative query systems aimed at providing easy, flexible, and intuitive access to information. Such systems are intended to facilitate retrieval from information repositories such as databases, libraries, and the Web. These repositories are typically equipped with standard query systems that are often inadequate for users. The focus of FQAS is the development of query systems that are more expressive, informative, cooperative, productive, and intuitive to use.

These proceedings contain contributions from invited speakers and 35 original papers out of more than 100 submissions, relating to the topic of users posing queries and systems producing answers. The papers cover the fields: database management, information retrieval, domain modeling, knowledge representation and ontologies, knowledge discovery and data mining, artificial intelligence, classical and non-classical logics, computational linguistics and natural language processing, multimedia information systems, and human-computer interaction, including reports of interesting applications. We wish to thank the contributors for their excellent papers and the referees, publisher, and sponsors for their effort. Special thanks to Serge Abiteboul, Elisa Bertino, and Yuzuru Tanaka who presented invited talks at the conference. We express our appreciation to the members of the advisory board, and members of the program committee. They made the success of FQAS 2004 possible.

The contributed papers were selected by the following program committee members: Sihem Amer-Yahia, Troels Andreasen, Boualem Benatallah, Djamal Benslimane, Elisa Bertino, Gloria Bordogna, Bernadette Bouchon-Meunier, Torben Brauner, Henrik Bulskov, Sylvie Calabretto, Henning Christiansen, Fabio Crestani, Juan Carlos Cubero, Ernesto Damiani, Rita De Caluwe, Guy De Tre, Hendrik Decker, Robert Demolombe, Marcin Detyniecki, Didier Dubois, Ronald R. Fagin, Rudolf Felix, Elena Ferrari, Jorgen Fischer Nilsson, Norbert Fuhr, Peter Ingwersen, Christian Jacquemin, Janusz Kacprzyk, Etienne Kerre, Rasmus Knappe, Don Kraft, Werasak Kurutach, Mounia Lalmas, Henrik L. Larsen, Christophe Marsala, Maria Jose Martin-Bautista, Saddaki Miyamoto, Amihai Motro, Noureddine Mouaddib, Fred Petry, Olivier Pivert, Olga Pons, Zbigniew Ras, Guillaume Raschia, Brigitte Safar, Michel Scholl, Dietmar Seipel, Andrzej

Skowron, Heiner Stuckenschmidt, Zahir Tari, Hanne Erdman Thomsen, Farouk Toumani, Peter Vojtas, Slawomir T. Wierzchon, Ronald R. Yager, Adnan Yazici, Slawomir Zadrozny, Djamel Zighed,

Additionally, we acknowledge the help in reviewing papers from: Aybar Acar, James Allen, Bernd Amann, Joachim Baumeister, Salima Benbernou, Patrick Bosc, Matilde Celma, Pierre-Antoine Champin, Jérôme Darmont, Martine De Cock, Fabien De Marchi, Susan Dumais, Elod Egyd-Zsigmond, Peter Gursky, Bernd Heumesser, Marbod Hopfner, Mieczyslaw Klopotek, Stanislav Krajci, K.L. Kwok, Monica Landoni, Manish Malhotra, Juan Miguel Medina, Henrik Nottelmann, Jean-Francois Omhover, Jean-Marc Petit, Daniel Rocacher, Khalid Saeed, André Schaefer, Stefan Schlobach, Mihaella Scuturici, Padmini Srinivasan, Anastasios Tombros, Dietrich Van der Weken, Keith van Rijsbergen, Abhinav Vora

March 2004

Henning Christiansen
Mohand-Said Hacid
Troels Andreasen
Henrik Legind Larsen

Table of Contents

Invited Papers

Knowledge Representation

Knowledge Discovery and Data Mining

Domain Modeling, Ontologies and Human Interaction

Computational Linguistics and Natural Language Processing

Web Services

Databases

Information Retrieval

Logic and Artificial Intelligence

Meme Media Architecture for the Reediting and Redistribution of Web Resources

Yuzuru Tanaka and Kimihito Ito

Meme Media Laboratory, Hokkaido University
Sapporo, 060-8628 Japan
{tanaka,itok}@meme.hokudai.ac.jp

Abstract. This paper reviews our R&D studies on meme media technologies and their application to Web resources. Meme media technologies make the Web work as a meme pool, where people can publish their intellectual resources as Web pages, access some Web pages to extract some of their portions as meme media objects through drag-and-drop operations, visually combine these meme media objects together with other meme media objects to compose new intellectual resources, and publish these resources again as Web pages. Such a visual composition through direct manipulation can define not only the layout of components, but also interoperations among these components.

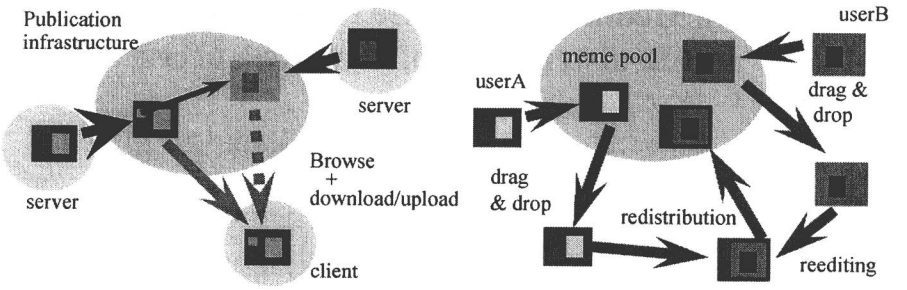
1 Introduction

During the last decade, we have observed the rapid accumulation of intellectual resources on the Web. These intellectual resources include not only multimedia documents, but also application tools running on the client side, and services provided by remote servers. Today, from the Web, you can almost obtain whatever information items, application tools, or services you may think of.

The publication and reuse of intellectual resources using the Web technologies can be characterized by the schematic model in Figure 1 (a). The Web publication uses a compound document representation of intellectual resources. Compound documents denote documents with embedded contents such as multimedia contents, visual application tools, and/or interactive services provided by servers. Such a compound document published on the Web defines a Web page. The model in Figure 1 (a) has no support for us to extract any portion of published Web pages, to combine them together for their local reuse, nor to publish the newly defined composite object as a new Web page. The composition here means not only textual combination but functional federation of embedded tools and services. We need some support to reedit and to redistribute Web contents for their further reuse.

It is widely recognized that a large portion of our paperwork consists of taking some portions of already existing documents, and rearranging their copies in different formats on different forms. Since the reediting is so fundamental in our daily information processing, personal computers introduced the copy-and-paste operation as the most fundamental operation. We need to make this operation applicable not only to multimedia documents but also to documents with embedded tools and services.

H. Christiansen et al. (Eds.): FQAS 2004, LNAI 3055, pp. 1–12, 2004.



(a)Current Web technologies (b)Meme media technologies

Fig. 1. The publication and reuse of intellectual resources

Figure 1 (b) shows a new model for the worldwide publication, reediting and redistribution of intellectual resources. As in the case of the Web, you can publish a set of your intellectual resources as a compound document into a worldwide publication repository. You can use a browser to view such documents published by other people. In addition to these operations, you can extract any portions of viewed documents as reusable components, combine them together to define a new compound document for your own use, and publish this new compound document into the repository for its reuse by other people. This new model of publishing, reediting and redistributing intellectual resources assumes that all these operations can be performed only through direct manipulation. Meme media technologies proposed by our group [1, 2, 3, 4], when applied to the Web, realize this new model, and make the Web work as a meme pool of intellectual resources. They provide the direct manipulation operations necessary for reediting and redistributing intellectual resources.

Web Service technologies can provide us with similar functions for the interoperation among Web applications [5, 6, 7, 8]. Web Service technologies enable us to interoperate services published over the Web. However, they assume that the API (Application Program Interface) library to access such a service is a priori provided by its server side. You need to write a program to interoperate more than one Web service. Meme media technologies, on the other hand, provide only the client-side direct manipulation operations for users to reedit intellectual resources embedded in Web pages, to define a new combination of them together with their interoperation, and to republish the result as a new Web page. In addition, meme media technologies are applicable not only to the Web, but also to local objects. Meme media can wrap any documents and tools, and make each of them work as interoperable meme media object. You can easily combine Web resources with local tools.

This paper shows how the meme media architecture is applied to the Web to make it work as meme pools. This makes the Web work as a shared repository not only for publishing intellectual resources, but also for their collaborative reediting.

2 Wrapping Web Contents as Meme Media Objects

2.1 Meme Media Architecture IntelligentPad

IntelligentPad is a two-dimensional representation meme media architecture. Its architecture can be roughly summarized as follows for our current purpose. Instead of directly dealing with component objects, IntelligentPad wraps each object with a standard pad wrapper and treats it as a pad. Each pad has both a standard user interface and a standard connection interface. The user interface of every pad has a card like view on the screen and a standard set of operations like 'move', 'resize', 'copy', 'paste', and 'peel'. As a connection interface, every pad provides a list of slots, and a standard set of messages 'set', 'gimme', and 'update'. Each pad defines one of its slots as its primary slot. Most pads allow users to change their primary slot assignments. You may paste a pad on another pad to define a parent-child relationship between these two pads. The former becomes a child of the latter. When you paste a pad on another, you can select one of the slots provided by the parent pad, and connect the primary slot of the child pad to this selected slot. The selected slot is called the connection slot. Using a 'set' message, each child pad can set the value of its primary slot to the connection slot of its parent pad. Using a 'gimme' message, each child pad can read the value of the connection slot of its parent pad, and update its primary slot with this value. Whenever a pad has a state change, it sends an 'update' message to each of its child pads to notify this state change. Whenever a pad receives an 'update' message, it sends a 'gimme' message to its parent pad. By pasting pads on another pad and specifying slot connections, you may easily define both a compound document layout and interoperations among these pads.

2.2 Extraction and Reediting of Web Contents

Web documents are defined in HTML format. An HTML view denotes an arbitrary HTML document portion represented in the HTML document format. The pad wrapper to wrap an arbitrary portion of a Web document specifies an arbitrary HTML view and renders any HTML document. We call this pad wrapper an HTMLviewPad. Its rendering function is implemented by wrapping a legacy Web browser Internet Explorer. The specification of an arbitrary HTML view over a given HTML document requires the capability of editing the internal representation of HTML documents, namely, DOM trees. The DOM tree representation allows you to identify any HTML-document portion, which corresponds to a DOM tree node, with its path expression such as /HTML[0]/BODY[0]/TABLE[0]/TR[1]/TD[1].

The definition of an HTML view consists of a source document specification, and a sequence of view editing operations. A source document specification uses the document URL. Its retrieval is performed by the function 'getHTML' in such a way as

doc = getHTML("http://www.abc.com/index.html", null).

The second parameter will be used to specify a request to the Web server at the retrieval time. Such requests include POST and GET. The retrieved document is kept in

DOM format. The editing of an HTML view is a sequence of DOM tree manipulation operations selected out of the followings:

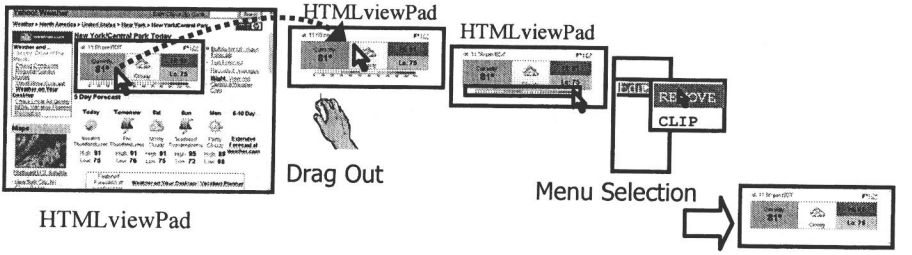
(1) EXTRACT : Delete all the nodes other than the sub tree with the specified node as its root.
(2) REMOVE : Delete the sub tree with the specified node as its root.
(3) INSERT : Insert a given DOM tree at the specified relative location of the specified node. You may select the relative location out of CHILD, PARENT, BEFORE, and AFTER.

An HTML view is specified as follows:

defined-view = *source-view.DOM-tree-operation*(*node*),

where source-view may be a Web document or another HTML document, and node is specified by its extended path expression. The following is an example view definition.

```
view1 = doc
        .EXTRACT("/HTML/BODY/TABLE[0]/")
        .EXTRACT("/TABLE[0]/TR[0]/")
        .REMOVE("/TR[0]/TD[1]/");
```



(a) EXTRACT operation (b) REMOVE operation

Fig. 2. Direct Manipulations for extracting and removing views

2.3 Direct Editing of HTML Views

Instead of specifying a path expression to identify a DOM tree node, we will make the HTMLviewPad to dynamically frame different extractable document portions for different mouse locations so that its user may move the mouse cursor around to see every extractable document portion. When the HTMLviewPad frames what you want to extract, you can drag the mouse to create another HTMLviewPad with this extracted document portion. The new HTMLviewPad renders the extracted DOM tree on itself. Figure 2 (a) shows an example extraction, which internally generates the following edit code.

```
doc = getHTML("http://www.abc.com/index.html", null);
view = doc
    .EXTRACT("/HTML/BODY/TABLE[0]/");
```