

V. S. SUBRAHMANIAN
SUSHIL JAJODIA (Eds.)

Multimedia Database Systems

Issues and
Research
Directions
多媒体数据库系统

Springer-Verlag
世界图书出版公司

V.S. Subrahmanian
Sushil Jajodia (Eds.)

Multimedia Database Systems

Issues and Research Directions

With 104 Figures and 9 Tables

792

Prof. V.S. Subrahmanian
University of Maryland
Computer Science Department
College Park, MD 20742
USA

Prof. Sushil Jajodia
George Mason University
Dept. of Information and Software
Systems Engineering
Fairfax, VA 22030
USA

Catalogue in Publication-Data applied for

ISBN 3-540-58710-1 Springer-Verlag Berlin Heidelberg New York

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Subrahmanian, V.S.: Multimedia database systems: issues and research directions/
V.S. Subrahmanian, Sushil Jajodia; . - Berlin; Heidelberg; New York; Barcelona;
Budapest; Hong Kong; London; Mailand; Paris; Santa Clara; Singapore; Tokyo:
Springer, 1996 (Artificial Intelligence) ISBN 3-540-58710-1 NE: Subrahmanian, V.S.:

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on micro-film or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

© Springer-Verlag Berlin Heidelberg 1996
Printed in Germany

The use of general descriptive names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

This reprint has been authorized by Springer-Verlag (Berlin/Heidelberg/New York) for sale in the People's Republic of China only and not for export therefrom.
Reprinted in China by Beijing World Publishing Corporation, 1998

Foreword

With the rapid growth in the use of computers to manipulate, process, and reason about multimedia data, the problem of how to store and retrieve such data is becoming increasingly important. Thus, although the field of multimedia database systems is only about 5 years old, it is rapidly becoming a focus for much excitement and research effort.

Multimedia database systems are intended to provide unified frameworks for requesting and integrating information in a wide variety of formats, such as audio and video data, document data, and image data. Such data often have special storage requirements that are closely coupled to the various kinds of devices that are used for recording and presenting the data, and for each form of data there are often multiple representations and multiple standards – all of which make the database integration task quite complex. Some of the problems include:

- what a multimedia database query means
- what kinds of languages to use for posing queries
- how to develop compilers for such languages
- how to develop indexing structures for storing media on ancillary devices
- data compression techniques
- how to present and author presentations based on user queries.

Although approaches are being developed for a number of these problems, they have often been ad hoc in nature, and there is a need to provide a principled theoretical foundation. To address that need, this book brings together a number of respected authors who are developing principled approaches to one or more aspects of the problems described above. It is the first book I know of that does so.

The editors of this book are eminently qualified for such a task. Sushil Jajodia is respected for his work on distributed databases, distributed heterogeneous databases, and database indexing. V. S. Subrahmanian is well known for his work on nonmonotonic reasoning, deductive databases and heterogeneous databases – and also on several different media systems: MACS (Media Abstraction Creation System), and AVIS (Advanced Video Information System), and FIST (Face Information System, currently under development). It has been a pleasure working with them, and I am pleased to have been able to facilitate in some small way the publication of this book.

Preface

With the advent of the information superhighway, a vast amount of data is currently available on the Internet. The concurrent advances in the areas of image, video, and audio capture, and the spectacular explosion of CD-ROM technology has led to a wide array of non-traditional forms of data being available across the network as well. Image data, video data, audio data, all perhaps stored in multiple, heterogeneous formats, traditionally form the "core" of what is known today as multimedia data.

Despite the proliferation of such forms of media data, as well as the proliferation of a number of commercially available tools to manipulate this data, relatively little work has been done on the principles of multimedia information systems. What common characteristics do all these different media-types have in common? Can these characteristics be exploited so as to provide a "common core" skeleton that can be used as a platform on which other multimedia applications can be built? If so, how can this be accomplished? These, and other questions arise in the context of such multimedia systems.

In this book, we bring together a collection of papers that address each of these questions, as well as a number of other related questions.

The first paper, by Marcus and Subrahmanian, provides a basic theoretical foundation for multimedia information systems that is independent of any given application. The authors identify core characteristics common to a variety of media sources. They then show that these core characteristics can be used to build indexing structures and query languages for media data. They argue that query processing can be used as a way of specifying media presentations.

The paper by Gudivada et al. studies a specific kind of multimedia information system – those dealing only with image data. The authors describe various kinds of operations inherent in such systems (e.g. retrieving objects based on shape similarity). They then provide a unified framework, called the AIR model, that treats all these different operations in a unified manner.

The paper by Arya et al. describes the design and implementation of the QBISM system for storing and manipulating medical images. In contrast to the paper of Gudivada et al., in this paper, the authors study issues of logical database design by including two special data types – VOLUME and REGION to represent spatial information.

In the paper by Sistla and Yu, the authors develop techniques for similarity based retrieval of pictures. Their paper is similar in spirit to that of Gudivada et al. – the difference is that whereas Gudivada et al. attempt to develop a unified data model, Sistla and Yu formalize the process of inexact matching between images and study the mathematical properties resulting from such a formalization.

The paper by Aref et al. studies a unique kind of multimedia data, viz. handwritten data. The authors have developed framework called *Ink* in which a set of handwritten notes may be represented, and queried. The authors describe their representation, their matching/querying algorithms, their implemented system, and the results of experiments based on their system.

In the same spirit as the papers by Gudivada et al. and Sistla and Yu, the issue of retrieval by similarity is studied by Jagadish. However, here, Jagadish develops algorithms to index databases that require retrievals by similarity. He does this by mapping an object (being searched for) as well as the corpus of objects (the database) into a proximity space – two objects are similar if they are near each other in this proximity space.

Belussi et al.'s paper addresses a slightly different query – in geographic information systems, users often want to ask queries of the form: "Find all objects that are as close to (resp. as far from) object *O* as possible". The authors develop ways of storing GIS data that make the execution of such queries very efficient. They develop a system called Snapshot that they have implemented.

The paper by Ghandeharizadeh addresses a slightly different problem. Once a query has been computed, and we know which video objects must be retrieved and presented to the user, we are still faced with the problem of actually doing so. This issue is further complicated by the fact that video-data must be retrieved from its storage device at a specific rate – if not, the system will exhibit "jitter" or "hiccups". Ghandeharizadeh studies how to present video objects without hiccups.

The paper by Ozden et al. has similar goals to that of Ghandeharizadeh – they too are interested in the storage and retrieval of continuous media data. They develop data structures and algorithms for continuous retrieval of video-data from disk, reducing latency time significantly. They develop algorithms for implementing, in digital disk-based systems, standard analog operations like fast-forward, rewind, etc.

The paper by Marcus revisits the paper by Marcus and Subrahmanian and shows that the query paradigm developed there – which uses a fragment of predicate logic – can just as well be expressed in SQL.

Cutler and Candan study different multimedia authoring systems available on the market, evaluating the pros and cons of each.

Finally, Kashyap et al. develop ideas on the storage of metadata for multimedia applications – in particular, they argue that metadata must be stored

at three levels, and that algorithms to manipulate the meta-data must traverse these levels.

The refereeing of the papers by Marcus and Subrahmanian, Jagadish, Ozden et al., Marcus, and Kashyap et al. was handled by Sushil Jajodia. The refereeing process for the other papers was handled by V.S. Subrahmanian. In addition, all but three papers (Ozden et al., Kashyap et al., and Jagadish) were discussed for several hours each in Subrahmanian's Multimedia Database Systems seminar course at the University of Maryland (Spring 1995). We are extremely grateful to those who generously contributed their time, reviewing papers for this book. Furthermore, we are grateful to the authors for their contributions, and for their patience in making revisions. Finally, we are grateful to Kasim Selcuk Candan for his extraordinary patience in helping to typeset the manuscript, and to Sabrina Islam for administrative assistance.

We would like to dedicate this book to our parents.

V.S. Subrahmanian
College Park, MD

Sushil Jajodia
Fairfax, VA

September 1995

Table of Contents

Towards a Theory of Multimedia Database Systems

Sherry Marcus and V.S. Subrahmanian 1

1. Introduction.....	1
2. Basic Ideas Underlying the Framework.....	3
3. Media Instances	4
3.1 The Clinton Example.....	5
3.2 Examples of Media-Instances	8
4. Indexing Structures and a Query Language for Multimedia Systems	12
4.1 Frame-Based Query Language	12
4.2 The Frame Data Structure	15
4.3 Query Processing Algorithms	21
4.4 Updates in Multimedia Databases.....	22
5. Multimedia Presentations	24
5.1 Generation of Media Events = Query Processing	24
5.2 Synchronization = Constraint Solving	27
5.3 Internal Synchronization	28
5.4 Media Buffers	28
6. Related Work	29
7. Conclusions	31

A Unified Approach to Data Modelling and Retrieval for a Class of Image Database Applications

Venkat N. Gudivada, Vijay V. Raghavan, and Kanonluk Vanapipat 37

1. Introduction.....	37
2. Approaches to Image Data Modeling	39
2.1 Terminology.....	40
2.2 Conventional Data Models	40
2.3 Image Processing/Graphics Systems with Database Functionality	41
2.4 Extended Conventional Data Models.....	42
2.5 Extensible Data Models.....	42
2.6 Other Data Models.....	43
3. Requirements Analysis of Application Areas	43
3.1 A Taxonomy for Image Attributes.....	43
3.2 A Taxonomy for Retrieval Types	46

XII Table of Contents

3.3 Art Galleries and Museums	48
3.4 Interior Design	48
3.5 Architectural Design	49
3.6 Real Estate Marketing	49
3.7 Face Information Retrieval	50
4. Logical Representations	51
5. Motivations for the Proposed Data Model	52
6. An Overview of AIR Framework	53
6.1 Data Model	53
6.2 The Proposed DBMS Architecture	57
7. Image Database Systems Based on AIR Model	58
8. Image Retrieval Applications Based on the Prototype Implementation of AIR Framework	61
8.1 Realtors Information System	61
8.2 Face Information Retrieval System	62
9. Research Issues in AIR Framework	64
9.1 Query Interface	64
9.2 Algorithms for RSC and RSS Queries	67
9.3 Relevance Feedback Modeling and Improving Retrieval Effectiveness	68
9.4 Elicitation of Semantic Attributes	69
10. Conclusions and Future Direction	70
A. Image Logical Structures	74

The QBISM Medical Image DBMS

*Manish Arya, William Cody, Christos Faloutsos, Joel Richardson, and
Arthur Toga*

1. Introduction	79
2. The Medical Application	81
2.1 Problem Definition	81
2.2 Data Characteristics	82
3. Logical Design	83
3.1 Data Types	83
3.2 Spatial Operations	83
3.3 Schema	84
3.4 Queries	85
4. Physical Database Design	86
4.1 Representation of a VOLUME	87
4.2 Representation of a REGION	88
4.3 Conclusions	89
5. System Issues	89
5.1 Starburst Extensions	89
5.2 System Architecture	90
6. Performance Experiments	93

6.1 Experimental Environment	93
6.2 Single-study Queries	94
6.3 Multi-study Queries	96
6.4 Results from the Performance Experiments	97
7. Conclusions and Future Work	97

Retrieval of Pictures Using Approximate Matching

<i>A. Prasad Sistla and Clement Yu</i>	101
1. Introduction	101
2. Picture Representation	102
3. User Interface	103
4. Computation of Similarity Values	105
4.1 Similarity Functions	105
4.2 Object Similarities	106
4.3 Similarities of Non-spatial Relationships	107
4.4 Spatial Similarity Functions	108
5. Conclusion	111

Ink as a First-Class Datatype in Multimedia Databases

<i>Walid G. Aref, Daniel Barbarà, and Daniel Lopresti</i>	113
1. Introduction	113
2. Ink as First-Class Data	114
2.1 Expressiveness of Ink	115
2.2 Approximate Ink Matching	116
3. Pictographic Naming	117
3.1 Motivation	118
3.2 A Pictographic Browser	120
3.3 The Window Algorithm	121
3.4 Hidden Markov Models	122
4. The ScriptSearch Algorithm	124
4.1 Definitions	125
4.2 Approaches to Searching Ink	126
4.3 Searching for Patterns in Noisy Text	128
4.4 The ScriptSearch Algorithm	129
4.5 Evaluation of ScriptSearch	132
4.6 Experimental Results	134
4.7 Discussion	136
5. Searching Large Databases	137
5.1 The HMM-Tree	137
5.2 The Handwritten Trie	149
5.3 Inter-character Strokes	160
5.4 Performance	160
6. Conclusions	160

Indexing for Retrieval by Similarity

<i>H. V. Jagadish</i>	165
1. Introduction	165
2. Shape Matching	166
2.1 Rectangular Shape Covers	167
2.2 Storage Structure	169
2.3 Queries	171
2.4 Approximate Match	172
2.5 An Example	177
2.6 Experiment	178
3. Word Matching	180
4. Discussion	181

Filtering Distance Queries in Image Retrieval

<i>A. Belussi, E. Bertino, A. Biavasco, and S. Rizzo</i>	185
1. Introduction	185
2. Spatial Access Methods and Image Retrieval	187
2.1 Query Processor	187
2.2 Image Objects and Spatial Predicates	189
3. Snapshot	191
3.1 Regular Grid with Locational Keys	192
3.2 Clustering Technique	194
3.3 Extensible Hashing	195
3.4 Organization of Snapshot	198
4. Filtering Metric Queries with Snapshot	201
4.1 Search Algorithm	202
4.2 Min Algorithm	205
5. Optimization of Spatial Queries	210
6. Conclusions and Future Work	211

Stream-based Versus Structured Video Objects: Issues, Solutions, and Challenges

<i>Shahram Ghandeharizadeh</i>	215
1. Introduction	215
2. Stream-based Presentation	217
2.1 Continuous Display	218
2.2 Pipelining to Minimize Latency Time	224
2.3 High Bandwidth Objects and Scalable Servers	225
2.4 Challenges	226
3. Structured Presentation	227
3.1 Atomic Object Layer	229
3.2 Composed Object Layer	231
3.3 Challenges	232
4. Conclusion	235

The Storage and Retrieval of Continuous Media Data*Banu Özden, Rajeev Rastogi, and Avi Silberschatz* 237

1. Introduction	237
2. Retrieving Continuous Media Data	238
3. Matrix-Based Allocation	240
3.1 Storage Allocation	241
3.2 Buffering	243
3.3 Repositioning	243
3.4 Implementation of VCR Operations	244
4. Variable Disk Transfer Rates	245
5. Horizontal Partitioning	247
5.1 Storage Allocation	248
5.2 Retrieval	250
6. Vertical Partitioning	250
6.1 Size of Buffers	252
6.2 Data Retrieval	255
7. Related Work	257
8. Research Issues	257
8.1 Load Balancing and Fault Tolerance Issues	257
8.2 Storage Issues	258
8.3 Data Retrieval Issues	259
9. Concluding Remarks	260

Querying Multimedia Databases in SQL*Sherry Marcus* 263

1. Introduction	263
2. Automobile Multimedia Database Example	265
3. Logical Query Language	269
4. Querying Multimedia Databases in SQL	270
5. Expressing User Requests in SQL	274
6. Conclusions	276

Multimedia Authoring Systems*Ross Cutler and Kasım Selçuk Candan* 279

1. Introduction	279
2. Underlying Technology	280
2.1 ODBC	280
2.2 OLE	281
2.3 DDE	281
2.4 DLL	281
2.5 MCI	282
3. Sample Application - "Find-Movie"	282
4. Multimedia Toolbox 3.0	283

5. IconAuthor 6.0	287
6. Director 4.0	289
7. MAS's and Current Technology	290
7.1 How to improve MAS's?	291
7.2 How to Benefit from MAS's in Multimedia Research	294
8. Conclusion	295

Metadata for Building the Multimedia Patch Quilt

<i>Vipul Kashyap, Kshitij Shah, and Amit Sheth</i>	297
--	-----

1. Introduction	297
2. Characterization of the Ontology	300
2.1 Terminological Commitments: Constructing an Ontology	301
2.2 Controlled Vocabulary for Digital Media	302
2.3 Better understanding of the query	304
2.4 Ontology Guided Extraction of Metadata	305
3. Construction and Design of Metadata	306
3.1 Classification of Metadata	307
3.2 Meta-correlation: The Key to Media-Independent Semantic Correlation	309
3.3 Extractors for Metadata	312
3.4 Storage of Metadata	314
4. Association of Digital Media Data with Metadata	315
4.1 Association of Metadata with Image Data	315
4.2 Association of Symbolic Descriptions with Image Data	316
4.3 Metadata for Multimedia Objects	316
5. Conclusion	317

Contributors	321
---------------------------	-----

Towards a Theory of Multimedia Database Systems

Sherry Marcus¹ and V.S. Subrahmanian²

¹ 21st Century Technologies, Inc. 1903 Ware Road, Falls Church, VA 22043.

E-mail: marcus@nego.umiacs.umd.edu

² Institute for Advanced Computer Studies, Institute for Systems Research
Department of Computer Science, University of Maryland, College Park, Maryland 20742.

E-mail: vs@cs.umd.edu

Summary. Though there are now numerous examples of multimedia systems in the commercial market, these systems have been developed primarily on a case-by-case basis. The large-scale development of such systems requires a principled characterization of multimedia systems which is independent of any single application. It requires a unified query language framework to access these different structures in a variety of ways. It requires algorithms that are provably correct in processing such queries and whose efficiency can be appropriately evaluated. In this paper, we develop a framework for characterizing multimedia information systems which builds on top of the implementations of individual media, and provides a logical query language that integrates such diverse media. We develop indexing structures and algorithms to process such queries and show that these algorithms are sound and complete and relatively efficient (polynomial-time). We show that the generation of media-events (i.e. generating different states of the different media concurrently) can be viewed as a query processing problem, and that synchronization can be viewed as constraint solving. This observation allows us to introduce the notion of a media presentation as a sequence of media-events that satisfy a sequence of queries. We believe this paper represents a first step towards the development of multimedia theory.

1. Introduction

Though numerous multimedia systems exist in today's booming software market, relatively little work has been done in addressing the following questions:

- What are multimedia database systems and how can they be formally defined so that they are independent of any specific application domain ?
- Can indexing structures for multimedia database systems be defined in a similar uniform, domain-independent manner ?
- Is it possible to uniformly define both query languages and access methods based on these indexing structures ?
- Is it possible to uniformly define the notion of an update in multimedia database systems and to efficiently accomplish such updates using the above-mentioned indexing structures ?
- What constitutes a multimedia presentation and can this be formally defined so that it is independent of any specific application domain ?

In this paper, we develop a set of initial solutions to all the above questions. We provide a formal theoretical framework within which the above questions can be expressed and answered.

The basic concepts characterizing a multimedia system are the following: first, we define the important concept of a **media-instance**. Intuitively, a media-instance (e.g. an instance of video) consists of a body of information (e.g. a set of video-clips) represented using some storage mechanism (e.g. a quadtree, or an R-tree or a bitmap) in some storage medium (e.g. video-tape), together with some functions and/or relations (e.g. next minute of video, or who appears in the video) expressing various aspects, features and/or properties of this media-instance. We show that media-instances can be used to represent a wide variety of data including documents, photographs, geographic information systems, bitmaps, object-oriented databases, and logic programs, to name a few.

Based on the notion of a media-instance, we define a **multimedia system** to be a set of such media-instances. Intuitively, the concatenation of the states of the different media instances in the multimedia system is a snapshot of the global state of the system at a given point in time. Thus, for instance, a multimedia system (at time t) may consist of a snapshot of a particular video-tape, a snapshot of a particular audio-tape, and segments of affiliated (electronic) documentation. In Section 4., we develop a **logical query language** that can be used to express queries requiring multimedia accesses. We show how various "intuitive" queries can be expressed within this language. Subsequently, we define an **indexing structure** to store multimedia systems. The elegant feature of our indexing structure is that it is completely independent of the type of medium being used – in particular, if we are given a *pre-existing* representation/implementation of some information in some medium, our method shows how various interesting aspects (called "features") of this information can be represented, and efficiently accessed. We show how queries expressed in our logical query language can be efficiently executed using this indexing structure.

Section 5. introduces the important notion of a media presentation based on the notion of a **media-event**. Intuitively, a media-event reflects the global state of the different media at a fixed point in time. For example, if, at time t , we have a picture of George Bush on the screen (i.e. video medium) and an audio-tape of George Bush saying X , then this is a media-event with the video-state being "George Bush" and the audio-state being "George Bush saying X ." A **media presentation** is a sequence of media-events. Intuitively, a media-presentation shows how the states of different media-instances change over time. One of the key results in this paper is that any query generates a set of media-events (i.e. those media-events that satisfy the query). Consequently, the problem of specifying a media-presentation can be achieved by specifying a sequence of queries. In other words,

Generation of Media Events = Query Processing.

Finally each media-event (i.e. a global state of the system) must be “on” for a certain period of time (e.g. the audio clip of Bush giving a speech must be “on” when the video shows him speaking). Furthermore, the next media-event must come on immediately upon the completion of the current media-event. We show that this process of synchronizing media-events to achieve a deadline may be viewed as a constraint solving problem, i.e.

Synchronization = Constraint Solving.

2. Basic Ideas Underlying the Framework

In this section, we will articulate the basic ideas behind our proposed multimedia information system architecture. For now, we will view a media-source as some, as yet unspecified, representation of information. Exactly how this information is stored physically, or represented conceptually, is completely independent of our framework, thus allowing our framework to be interface with most existing media that we know of.

Suppose $\mathcal{M}$ is a medium and this medium has several “states” representing different bodies of knowledge expressed in that medium – associated with this data is a set of “features” – these capture the salient aspects and objects of importance in that data. In addition, there is logically specified information describing relationships and/or properties between features occurring in a given state. These relationships between features are encoded as a logic program. Last, but not least, when a given medium can assume a multiplicity of states, we assume that there is a corpus of state-transition functions that allow us to smoothly move from one state to another. These are encoded as “inter-state” relationships, specifying relations existing between states taken as a whole. As the implementation of these inter-state transition functions is dependent on the medium, we will assume that there is an existing implementation of these transition functions. As we make no assumptions on this implementation, this poses no restrictions. Figure 2.1 shows the overall architecture for multimedia information systems.

The ideas discussed thus far are studied in detail in Section 4. where we develop a query language to integrate information across these multiple media sources and express queries, and where we develop access structures to efficiently execute these queries.

All the aspects described thus far are independent of time and are relatively static. In real-life multimedia systems, time plays a critical role. For instance, a query pertaining to audio-information may need to be synchronized with a query pertaining to video-information, so that the *presentation* of the answers to these queries have a coherent audio-visual impact. Hence, the data structures used to represent information in the individual media (which so far,

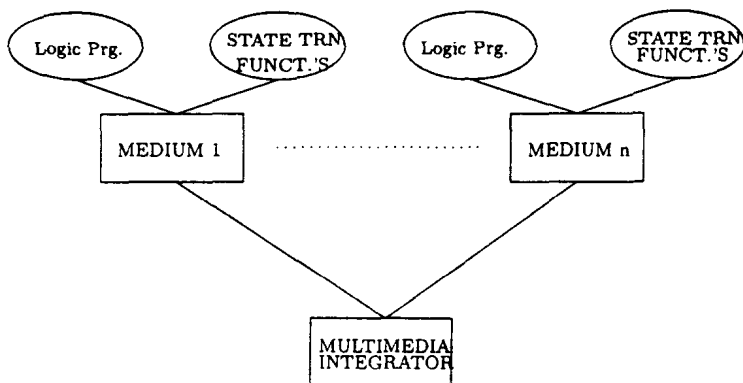


Fig. 2.1. Multimedia Information System Architecture

has been left completely unspecified) must satisfy certain efficiency requirements. We will show that by and large, these requirements can be clearly and concisely expressed as constraints over a given domain, and that based on the design criteria, index structures to organize information within a medium can be efficiently designed.

3. Media Instances

In this section, we formally define the notion of a media-instance, and show how it can be used to represent a wide variety of data stored on different kinds of media. Intuitively, a medium (such as video) may have data stored on it in many formats (e.g. raster, bitmap, vhs_format, pal, secam, etc.). Thus, raster is an example of an instance of the medium video because video information may be stored in raster format. However, in addition to just storing information, media instances, as defined below contain information on how to access and manipulate that information.

A *media-instance* is a 7-tuple $mi = (ST, fe, \lambda, \mathcal{R}, \mathcal{F}, Var_1, Var_2)$ where ST is a set of objects called *states*, fe is a set of objects called *features*, λ is a map from S to 2^{fe} , Var_1 is a set of objects called *state variables* ranging over states, Var_2 is a set of objects called *feature variables* ranging over features, $\mathcal{R}$ is a set of *inter-state relations*, i.e. relations (of possibly different arities) on the set ST , and $\mathcal{F}$ is a set of *feature-state relations*. Each relation in $\mathcal{F}$ is a subset of $fe^i \times ST$ where $i \geq 1$.