Panel on “Past and Future of Computer Science Theory”

(Discussion Paper)

Ronald Fagin[1], Georg Gottlob[2], Christos Papadimitriou[3], Moshe Vardi[4], Giorgio Ausiello[5], Maurizio Lenzerini[5], Luigi Palopoli[6], Domenico Saccà[6] and Francesco Scarcello[6]

[1] *IBM Research - Almaden, San Jose, CA 95120-6099, USA*
[2] *Computer Science Department, University of Oxford, OX1 3QD, UK*
[3] *Columbia University, New York, NY 10027, USA*
[4] *Computer Science Department, Rice University, Houston, TX 77005, USA*
[5] *DIAG Department, Sapienza University of Rome, 00185 Rome, Italy*
[6] *DIMES Department, University of Calabria, 87036 Rende, Italy*

Abstract

The twenty-ninth edition of the SEBD (Italian Symposium on Advanced Database Systems), held on 5-9 September 2021 in Pizzo (Calabria Region, Italy), included a joint seminar on “Reminiscence of TIDB 1981” with invited talks given by some of the participants to the Advanced Seminar on Theoretical Issues in Databases (TIDB), which took place in the same region exactly forty years earlier. The joint seminar was concluded by a Panel on “The Past and the Future of Computer Science Theory” with the participation of four distinguished computer science theorists (Ronald Fagin, Georg Gottlob, Christos Papadimitriou and Moshe Vardi), who were interviewed by Giorgio Ausiello, Maurizio Lenzerini, Luigi Palopoli, Domenico Saccà and Francesco Scarcello. This paper reports the summaries of the four interviews.

Keywords

Computer Science Theory, Database Theory

1. Introduction

Forty years ago, on September 13-19 of 1981, Grand Hotel San Michele (a fascinating resort in the Calabria’s Tyrrhenian coast in Southern Italy) hosted the Advanced Seminar on Theoretical Issues in Databases (TIDB’81), organized by Giorgio Ausiello, François Bancilhon, Domenico

SEBD 2021: The 29th Italian Symposium on Advanced Database Systems, September 5-9, 2021, Pizzo Calabro (VV), Italy

fagin@us.ibm.com (R. Fagin); georg.gottlob@cs.ox.ac.uk (G. Gottlob); christos@columbia.edu (C. Papadimitriou); vardi@cs.rice.edu (M. Vardi); ausiello@diag.uniroma1.it (G. Ausiello); lenzerini@diag.uniroma1.it (M. Lenzerini); luigi.palopoli@unical.it (L. Palopoli); domenico.sacca@unical.it (D. Saccà); francesco.Scarcello@unical.it (F. Scarcello)
0000-0002-7374-0347 (R. Fagin); 0000-0002-2353-5230 (G. Gottlob); 0000-0001-8084-0445 (C. Papadimitriou); 0000-0002-0661-5773 (M. Vardi); 0000-0003-1345-5007 (G. Ausiello); 0000-0003-2875-6187 (M. Lenzerini); 0000-0003-4915-5137 (L. Palopoli); 0000-0003-3584-5372 (D. Saccà); 0000-0001-7765-1563 (F. Scarcello)

Saccà and Nicolas Spyratos. In the early 1980s, database theory was at the early stage and TIDB'81 was one of the few pioneering initiatives organized in Europe.

TIDB'81 program included invited talks by outstanding scientists such as: Catriel Beeri, Ron Fagin, Seymour Ginsburg, Bob Kowalski, Witold Lipski, John Mylopoulos, Jean-Marie Nicolas, Christos Papadimitriou, Jan Paredaens, Yehoshua Sagiv, Mihalis Yannakakis, Carlo Zaniolo. Besides the main lectures, short talks were also given by a young (at that time) Italian scientist Carlo Batini and two promising postdocs Maurizio Lenzerini and Paolo Atzeni. More details on the seminar, including a pleasant description of the warm social atmosphere of the event, can be found in [1]. This book mentions, among other things, what Christos Papadimitriou said about TIDB'81 in the report written for the EATCS Bulletin: *There was an extensive mosaic of topics covered, reflecting the state of wild fermentation in which database theory appears to be. Relational theory, concurrency control, and logic in databases were three areas that were represented quite heavily in the list of speakers. Voices of the null value problem, schema design theory, the operational approach, conceptual modeling, and software engineering issues, were also heard.*

In 1984, a second edition of TIDB was held in Benodet (on the Brittany coast in North-Western France) and this edition too met with widespread participation and increased scientific interest. Therefore, a small group of people (Serge Abiteboul, Paolo Atzeni, Giorgio Ausiello, Catriel Beeri, Jan Paredaens, Domenico Sacca, Nicholas Spyratos) felt that there was a need to have a regular international forum for European researchers in database theory, following the format of the ACM Symposium on Principles of Database Systems (PODS), organized annually in the US starting from 1982. Then, the International Conference on Database Theory (ICDT) was set up as the successor of TIDB seminars and its first edition was held in Rome in 1986. At the 2011 celebration of the 30th anniversary of PODS, two events were remembered as progenitors of PODS and ICDT and for contributing to the birth of the field of database theory: the XP series of Workshops on Database Theory (XP1 took place at SUNY Stony Brook in 1980 and XP2 took place at Pennsylvania State University in 1981) and the TIDB Seminars.

Forty years later, on 9 September 2021, in the same Calabria's Tyrrhenian coast that hosted TIDB'81, a seminar on "Reminiscence of TIDB 1981" has been organized in connection with the twenty-ninth edition of the SEBD (the Italian Symposium on Advanced Database, the major annual event of the Italian database research community), held in Pizzo Calabro on 5-9 September 2021. The joint seminar on "Reminiscence of TIDB 1981" included invited talks given by the four organizers and two participants (Paolo Atzeni and Carlo Zaniolo) of TIDB'81 and a Panel on "The Past and the Future of Computer Science Theory". The panelists were four distinguished computer science theorists (Ronald Fagin, Georg Gottlob, Christos Papadimitriou and Moshe Vardi), who were interviewed by Giorgio Ausiello, Maurizio Lenzerini, Luigi Palopoli, Domenico Saccà and Francesco Scarcello.

The next four sections present the panel interview summaries in a question/answer format. The questions to each panelist were of three types: (1) his remembrance of database theory at the time of TIDB'81, (2) his major achievements in database and computer science theory from the late eighties on, and (3) his opinion on the future directions of database theory and, more in general, on the role that computer science theory will play in the next years.

2. Questions to Ronald Fagin

Ronald Fagin was interviewed by Luigi Palopoli and Domenico Saccà [2]. Ron Fagin is an IBM fellow at IBM Almaden research center. He is a distinguished scientist who provided fundamental contributions in several areas of theoretical computer science including database theory, logic in computer science, data exchange, reasoning about knowledge, finite model theory, information retrieval and indexing techniques. He is a member of the National Academy of Sciences, National Academy of Engineering, and American Academy of Arts and Sciences. He is an ACM Fellow and an IEEE Fellow. He obtained the Gödel Prize, the IEEE W. Wallace McDowell Award, the IEEE Technical Achievement Award, the ACM SIGMOD Edgar F. Codd Innovations Award, the Alonzo Church Award for Logic and Computation, several IBM Outstanding Innovation Awards and IBM Outstanding Technical Achievement Awards. Also, he received a Docteur Honoris Causa from the University of Paris, and a Laurea Honoris Causa from the University of Calabria.

1. *Ron, what are your reminiscence of discussions at TIDB'81 and, more in general, about the relevant topics you and other theoreticians used to investigate in the late seventies and early eighties?* (*By the way, we remember you working on Cetraro beach together with Marco Casanova and Christos Papadimitriou on some complexity issues of the interaction of inclusion database dependencies with functional dependencies* [3]).

   RF: When, in 1975, I moved from IBM Yorktown Heights to Almaden, I asked myself whom would be interesting to work with. And, actually, there were many interesting people around, but the one that stood out in my mind was Ted Codd. So I started to work with him on relational theory and my very first paper in this area was about data dependencies and entitled *Functional dependencies in relational databases and propositional logics* [4]. An important related topic I also addressed at TIDB was about acyclic hypergraphs and their relationships to database schemes. What is the correct notion of acyclicity for an hypergraph was not at all obvious at that time (as it was instead for ordinary graphs). I worked on this with Ullman, Vardi, Yannakakis, Maier and Mendelzon and we defined relevant notions and proved interesting properties about database schemes associated to acyclic hyergraphs [5, 6].

2. *Would you mind to tell us about the most relevant developments of your research after that pioneering period?.*

   RF: The first topic I would cite is reasoning about knowledge. I worked on that with Halpern, Moses and Vardi and, actually, the only book I wrote is about this very topic [7]. The basic questions in this context arise when trying to make sense of sentences like "*someone knows that someone else knows that a third party knows this or that*" and there are applications all over the places for this kind of reasoning and analysis: besides computer science, economics and political science and other areas where these kinds of questions naturally arise very often.
   Talking about data exchange, I would say that this topic is very fun. I worked on that with Kolaitis, Popa, Miller and Tan. There was a project called Clio on data exchange about moving data from one format to another and, after sitting for one year in meeting rooms, we decided to simply face the problem from scratch, studying it from a purely

theoretical point of view. We did that and we got some very interesting results that convinced practical people to re-implement their systems and our developments are now part of data exchange products. By the way, we won the Alonso Church award for the work on data exchange [8, 9]: this was a one time episode in which not only computer science borrows notions and results from logics, but also logics is fertilized by computer science - in this case with the notion of second-order dependency.
Another issue I would mention involved Laura Haas when she was on the Garlic project at IBM. Garlic was intended to be designed as a middleware software sitting on top of DB2 for relational data and QBIC for images in order to enable integrated querying of both sources. And Laura asked me for efficient ways to combine answers from the two domains, which clearly have very different data formats. So I developed an approach based on fuzzy logics and she seemed to be quite happy with that, but then she pointed out that straightforward ways to implement the approach would have required linear time, that was unaffordable for the huge data repositories handled by Garlic. Therefore, I developed a new algorithms that turned out to be very efficient in practice (nowadays known as the Fagin's algorithm) [10]. Few years later, together with Lotem and Naor, I developed a nice improvement of the Fagin's algorithm, known as the *threshold algorithm* [11], characterized by the so called instance optimality, which means to be optimal for every single input. This result took us to obtain the Godel Price, and that was the first database paper to win this price. Other issues I worked on producing interesting results are entity linking, differential back-ups and extensible hashing, which are very relevant in databases.

3. *Within the ample spectrum of your research contributions to database theory and, more in general, to theoretical computer science, which ones do you consider will have the most durable influence in the long run? Furthermore, which topics do you think will be relevant and hot in database theory in the years to come?.*

   RF: It is difficult to answer the first part of your question: it is like asking someone to choose among his children! Anyway, I would say that many of them are significant and retain their relevance to date, as in fact many of them are currently implemented in systems and products. As for the second part of the question, I think AI will be the big hot topic in the future also for database people. For instance, I am currently involved in working on real valued logics (a kind analogous to fuzzy logics) which is relevant in managing huge logical neural networks and optimizing them. So, again, AI is a very large area hosting relevant future research directions.

4. *What is your vision on the role of Computer Science Theory in the future?*
   RF: I am a very strong believer of the importance of theory and mathematics in computer science, to formally prove things work, to prove how long it takes for an algorithm to run or how much space it is needed. I am definitely sure that mathematics and theory will be with us forever, because we need precision, we need precise definitions, we need to prove results about how systems behave. So, I am very optimistic about the future of the role of theorists in computer science.
5. *So, can we tell the young boys and girls in the audience that they should work on theory because the future is there?*

RF: Yes, for sure!

3. Questions to Georg Gottlob

Georg Gottlob was interviewed by Francesco Scarcello and Domenico Saccà [12]. Georg Gottlob is a Professor at the Oxford University Department of Computer Science, and a Fellow of St John's College; he is Adjunct Professor at the Technische Universität in Vienna. He provided many influential contributions, by developing a common core to the underlying principles of artificial intelligence and databases, and by solving open problems in computational logic, non-monotonic reasoning and database theory. Ha was elected as a member of the Royal Society in 2010, and was recipient of many awards, such as the Wittgenstein Award, which is the highest funded Austrian award for scientific achievements, and the ECCAI fellowship. He also received a number of honorary lauree and doctorates from the University of Klagenfurt, the University of Vienna, and the University of Calabria.

1. *What about Database Theory in the 80's? You are too young at the time of TIDB '81 – how do you have later heard of it?*

   GG I knew about this pioneering workshop while I was a post-doc at Politecnico of Milano, where I also first met Mimmo Saccà, who was giving a seminar about his recent contributions on hypergraphs of functional dependencies [13, 14]. Before coming to Italy, I did joint work with Alexander Leitsch on the efficiency of *subsumption algorithms*, which find many applications in computer science and, in particular, in database theory [15]. I also studied the relationship between subsumption and implication, by identifying classes where the two notions actually coincide. I believe that very important and lively contributions in that period were about *data dependencies*, though an important scientist at PODS 1987 [16] looked at such a subject as the "last gasps of the dying swans" (which gave me quite a scare, as a young researcher just presenting a paper about covers of embedded functional dependencies! [17]) It was also the time of *deductive databases*, where logic programming was applied to databases. I enjoyed working with Stefano Ceri and Letizia Tanca at a book on Datalog covering many topics, from semantics to implementation issues and efficiency [18].

2. *Let us talk about relevant and influential theory topics and results in Database Theory and Artificial Intelligence, two fields where you played a prominent role and provided outstanding contributions.*

   GG Among the many interesting topics relating Database Theory and Artificial Intelligence, I intensively worked on the *structural properties* of problems which allow us of identify tractable classes of problems that are intractable, in general. With Francesco Scarcello and Nicola Leone, we worked on hypergraphs characterizing conjunctive queries in database and Constraint Satisfaction Problems (CSPs) in Artificial Intelligence (as well as many other problems related to homomorphisms between relational structures) [19]. We found that the complexity class LOGCFL plays an important role for such a (hyper)-tree shaped problems. In particular, it turned out that answering an acyclic query is not only feasible in polynomial-time, but also LOGCFL-complete, and hence parallelizable. The same holds

for those queries (or CSPs) having a small degree of cyclicity, more precisely, those queries having a small width *hypertree decomposition* [20].
Another interesting line of research was about the *Complexity and Expressive Power* of Logical Reasoning, where we look not only at the complexity of reasoning problems, but also on the possible ability of a given language to express all problems in a certain complexity classes. For instance, it has been shown that disjunctive Datalog is able to express all problems at the second level of the polynomial hierarchy [21], and FO logic with an oracle in NP is able to express all problems in L^{NP} [22]. Another interesting example is that Monadic Datalog over trees expresses exactly full Monadic Second Order Logic over trees, and the complexity of evaluation is $|Data| \times |Program|$ [23].
These research activities lead to concrete applications and even to the birth of spinoff companies, such as DLVsystem, Lixto, Wrapidity, and DeepReason.ai.

3. *Our last question is about future: what you see as the emerging topics in database and AI, and which role they may play in computer science theory in the next years?*

   A first interesting topic is about using database theory and techniques for Machine Learning and, in particular, for Deep Learning. Usually, in these applications everything goes into a very big flat relation, where we miss information on possible relationships among fragments of data. A second hot topic is about how we can combine reasoning with machine learning: there are human-generated rules that allow us as humans to learn and reason at an abstract and high level that deep learning systems cannot achieve, while they are instead exceptionally good at recognizing tasks and objects. Symbolic and sub-symbolic systems should work together, which needs perhaps adding some (new) form of probabilistic features to logical frameworks such as Datalog.

4. Questions to Christos Papadimitriou

Christos Papadimitriou was interviewed by Giorgio Ausiello and Domenico Saccà [24]. A Columbia University Professor, Christos is one of world's leading computer science theorists and is best known for his fundamental work in computational complexity. Using algorithmic lens, Christos has also explored other fields such as biology and the theory of evolution, economics and game theory, artificial intelligence, robotics, networks and the Internet. Recently he is attracted by the study the algorithms aspects of the brain. Christos has been the recipient of many prestigious scientific awards such as Awards: the Knuth Prize, IEEE's John von Neumann Medal, the EATCS Award, the IEEE Computer Society Charles Babbage Award, the Gödel Prize.

1. *Christos, we remember that at TIDB '81 you made a strong statement: "if you got an IT problem, look at yellow pages for CS theoreticians?" Again at Kyoto ICALP in 2015, where you gave a talk to celebrate 40 years of the journal Theoretical Computer Science, you said "In the Eighties theoreticians owned Computer Science". Did you mean that theoreticians were mastering all of CS?*

   CP This is my interpretation of history: In the Seventies and Eighties theory had a core program. Thanks to the results of Cook and Karp and others, we realized that we could study concrete CS problems and at the same time do good mathematics. I am only talking about the area of algorithms and complexity. I know that those were great decades also for track B[1], and even though for me theory is algorithms and complexity, back then you had to know about track B. For any emerging technology that was developed, we had to provide formalization and mathematical formulation. Databases was one of these areas – perhaps the most glorious. So we were not the masters of CS, we were its servants. I remember that when memories with magnetic bubbles came about, Mihalis Yannakakis and I were wandering how to treat them in theoretical terms — thankfully they went away quickly. Then in the Nineties every sub-field of CS started to develop its own theoretical framework and to have its own dedicated theoreticians, and by now there are no generalist theoreticians addressing all possible fields of CS. So, we started looking outside CS.

2. *Christos, more recently, in the Nineties you developed new very interesting fields of theoretical computer science. In particular can you tell us something about how you got interested in game theory?*

   CP In 1996 when I moved from San Diego to Berkeley, the Internet buzz was everywhere around me, and I started to think about the Internet with Prabakar Raghavan and to address issues related to it [27, 28] such as web search etc. The Internet cannot be called an artifact, it has not been created by a designer or company. The Internet has emerged from the interactions of many, thousands or millions of entities and users, each with their own incentives. This changed the rules of computation. It is in this context that I started to look at distributed mechanisms to reach a goal, involving agents who have different interests. Game theory and mechanism design and economic thinking became very important.

3. *Again in your Kyoto talk we remember that you mentioned your research work regarding evolution. You were trying to understand what evolution tries to optimize. Have you made any progress in this area?*

   CP Yes, actually we developed this subject a few years ago. When CS theoreticians start to look at new domains of science, they apply their computational lens. A lot of other areas can benefit of their ability to dissect the computational aspects implicit in natural or social systems. Together with Erick Chastain, Adi Livnat and Umesh Vazirani I wrote a paper: 'Algorithms, games and evolution' [29]. We reframed the equations of evolution in terms of convex programming and duality theory, and in terms of no-regret learning. We came to the conclusion that, from the point of view of genes, evolution optimizes a sort of convex combination of fitness and genetic diversity.

4. *Among the emerging field of theoretical computer science neural networks and deep learning have now assumed an extremely important role. What is your view of this trend?*

   CP Deep nets are an admirable advance, a fascinating chapter of computation. But my impression is that a real breakthrough happened after 2015 when deep learning started to be applied to natural language processing. Systems like Transformers or GPT3 have marked a

[1] This term comes from the Handbook on Theoretical Computer Science, which had two volumes: A was for algorithms and complexity [25], and B was for logic and semantics [26]

real revolution compared to neural networks used for image classification. Natural language is the means whereby people exchanged and recorded their thoughts while interacting with the world through their senses, over the centuries. I think it makes much more sense to train deep nets on Wikipedia or the library of Congress for the key to intelligence, rather than on images.

5. *As you certainly know we are now in a period in which Bibliometrics has become a major instrument for evaluating research activity. Nowadays many scientists are committed to write at least two papers per month as much as a war reporter! Can a theoretician prove two theorems per month? Or can a scientist conduct an experiment in 15 days? What is your view about this unpleasant situation?*

   CP Once, only biologists or physicists with big labs could write so many papers based on the results of their lab's experimental work. And in the old days a mathematician had to spend a long time trying to gather the relevant literature, and this has now become much easier. But you are right, the scientific production has become like an arms race, out of control, pushing researchers to write more and more papers. The time to produce a paper has shortened and researchers tend to write their results fresh off their brains. You can no longer read all papers that are published in some domain, you must rely on students and reviews to identify the important work in the deluge. I am at a point of my career where it only makes sense to focus on the one thing you believe is most important (and for me this is how the brain makes language), and yet I end up publishing more papers because I have students and collaborators, and of course because science is endless and problems are exciting and it is fun to indulge.

5. Questions to Moshe Vardi

Moshe Vardi was interviewed by Maurizio Lenzerini, Domenico Saccà and Francesco Scarcello [30]. Moshe Vardi, Rice University Professor and the Karen Ostrum George Distinguished Service Professor of Computational Engineering, is one of the most prominent Computer Scientists, who provided outstanding contributions in several aspects of Computer Science Theory, including Data Management, Automata Theory, Logic in Computer Science, Automated Reasoning, and Design Specification and Verification.

1. *Moshe, you did not attend TIDB'81, but you were already part of the Database community at that time. Which are the most important accomplishments of Database Theory in the 80s?*

   MV: To answer a question like that, we have to agree on the meaning of "important", "relevant", "significant", etc. These are qualitative adjectives, which are very difficult to precisely define in the context of scientific work. I recently did an exercise: I picked up a bunch of database theoreticians, namely Catriel Beeri, Jeff Ullman, Ron Fagin, Christos Papadimitriou, Mihalis Yannakakis and myself, and I checked how many publications from the 80s with more than 500 citations they have. The result was somehow surprising: not many such papers got more than 500 citations. Is it a sign that not many research results in

Database theory have had an impact? Very difficult to say!

2. *We agree that an objective answer to the question of relevance of research results is very difficult to give. So, let us switch to a different flavor of the question: what is your personal opinion about significant research results from Database Theory in the 80s?*

   MV: A first topic where foundational results have been produced in the 80s is the theory of data dependencies [31]. Dependencies are special forms of logical assertions that have several applications in the context of Data Management. Results on dependencies had an impact on various issues, such as database design, reasoning on data schemas, data quality and others. They also proved extremely important in both data exchange and, indirectly, Datalog. A second topic where fundamental research has been carried in the 80s is logic programming, which is the paradigm at the basis of the Datalog query language, as well as several rule-based formalisms, including existential rules and ontology languages. Indeed, the idea of rules seems very fundamental both from the conceptual point of view, and from the point of view of applications. For example, the impact of Datalog has been prominent in areas such as network programming, program analysis and web information extraction, maybe even more than in database systems. A third topic extensively studied in the 80s is query processing and analysis, especially the study of structural properties of queries, such as acyclicity or hypertree decomposition. Such investigations have been important for designing specialized algorithms for query answering, and singling out cases where tractability is guaranteed, taking into account the difference between data and query complexity [32].

3. *You are also recognized for your outstanding work as Editor in Chief of "Communication of the ACM". How was this experience?*

   MV: It has been a great experience, especially with respect to the goal of achieving a broad perspective of our discipline, and thinking about its potentials in addressing critical problems of our society. When you have the possibility of having a large view on the research carried out on a certain discipline, on one hand you can appreciate the variety of topics investigated by the various groups, and on the other hand you realize that sometimes the research problems addressed by the scientists do not really match with the big questions that our society is facing. I really believe that many such problems could find relevant answers with the help of Computer Science.

4. *Let us end with an eye to the future. What do you think will be the future of theory in the context of Data Management?*

   MV: I would mention two issues that I think will be prominent in the near future. The first has to do with reconciling the two aspects of Data Management that will be crucial in the data-driven society that we are experiencing, namely data analytics and data governance. Indeed, while the recent success of machine learning and data mining have stressed the role of data analysis in extracting information and knowledge from data, many observers point out the growing need of data governance, i.e. the development of effective methods for collecting and organizing data, controlling their quality and managing important properties such as completeness, consistency, fairness, and privacy. The second issue concerns Computer Science in general: while the theory of computational complexity has shown its mathematical beauty and its importance in algorithm analysis, it is now clear that it is not sufficient for explaining the behaviour of algorithms in practice. This aspect is particular important in the light of the roles played by algorithms in our society.

References

[1] G. Ausiello, *The Making of a New Science - A Personal Journey Through the Early Years of Theoretical Computer Science*, Springer, 2018. URL: https://doi.org/10.1007/978-3-319-62680-2. doi:10.1007/978-3-319-62680-2.

[2] Panel on Past and Future of Computer Science Theory, *Ronald Fagin's interview,* by Luigi Palopoli and Domenico Saccà, 2021. URL: https://drive.google.com/file/d/126EXLSpmKxV724l7LaX9qrZZmhzdG3YQ/view?usp=sharing, accessed: 2021-09-09.

[3] M. A. Casanova, R. Fagin, C. H. Papadimitriou, Inclusion dependencies and their interaction with functional dependencies, J. Comput. Syst. Sci. 28 (1984) 29–59. URL: https://doi.org/10.1016/0022-0000(84)90075-8. doi:10.1016/0022-0000(84)90075-8.

[4] R. Fagin, Functional dependencies in a relational data base and propositional logic, IBM J. Res. Dev. 21 (1977) 543–544. URL: https://doi.org/10.1147/rd.216.0534. doi:10.1147/rd.216.0534.

[5] C. Beeri, R. Fagin, D. Maier, A. O. Mendelzon, J. D. Ullman, M. Yannakakis, Properties of acyclic database schemes, in: Proceedings of the 13th Annual ACM Symposium on Theory of Computing, May 11-13, 1981, Milwaukee, Wisconsin, USA, ACM, 1981, pp. 355–362. URL: https://doi.org/10.1145/800076.802489. doi:10.1145/800076.802489.

[6] C. Beeri, R. Fagin, D. Maier, M. Yannakakis, On the desirability of acyclic database schemes, J. ACM 30 (1983) 479–513. URL: https://doi.org/10.1145/2402.322389. doi:10.1145/2402.322389.

[7] R. Fagin, J. Y. Halpern, Y. Moses, M. Y. Vardi, Reasoning About Knowledge, MIT Press, 1995. URL: https://doi.org/10.7551/mitpress/5803.001.0001. doi:10.7551/mitpress/5803.001.0001.

[8] R. Fagin, P. G. Kolaitis, R. J. Miller, L. Popa, Data exchange: Semantics and query answering, in: D. Calvanese, M. Lenzerini, R. Motwani (Eds.), Database Theory - ICDT 2003, 9th International Conference, Siena, Italy, January 8-10, 2003, Proceedings, volume 2572 of *Lecture Notes in Computer Science*, Springer, 2003, pp. 207–224. URL: https://doi.org/10.1007/3-540-36285-1_14. doi:10.1007/3-540-36285-1_14.

[9] R. Fagin, P. G. Kolaitis, L. Popa, W. C. Tan, Composing schema mappings: Second-order dependencies to the rescue, in: C. Beeri, A. Deutsch (Eds.), Proceedings of the Twenty-third ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 14-16, 2004, Paris, France, ACM, 2004, pp. 83–94. URL: https://doi.org/10.1145/1055558.1055572. doi:10.1145/1055558.1055572.

[10] R. J. Miller, M. A. Hernández, L. M. Haas, L. Yan, C. T. H. Ho, R. Fagin, L. Popa, The clio project: Managing heterogeneity, SIGMOD Rec. 30 (2001) 78–83. URL: https://doi.org/10.1145/373626.373713. doi:10.1145/373626.373713.

[11] R. Fagin, A. Lotem, M. Naor, Optimal aggregation algorithms for middleware, J. Comput. Syst. Sci. 66 (2003) 614–656. URL: https://doi.org/10.1016/S0022-0000(03)00026-6. doi:10.1016/S0022-0000(03)00026-6.

[12] Panel on Past and Future of Computer Science Theory, *Georg Gottlob's interview*, by Francesco Scarcello and Domenico Saccà, 2021. URL: https://drive.google.com/file/d/11llc8ou8L_12CEj7SM3bPpk_BhQBZf4_/view?usp=sharing, accessed: 2021-09-09.

[13] G. Ausiello, A. D'Atri, D. Saccà, Graph algorithms for functional dependency manipulation, J. ACM 30 (1983) 752–766. URL: https://doi.org/10.1145/2157.322404. doi:10.1145/2157.322404.

[14] D. Saccà, Closures of database hypergraphs, J. ACM 32 (1985) 774–803. URL: https://doi.org/10.1145/4221.4997. doi:10.1145/4221.4997.

[15] G. Gottlob, A. Leitsch, On the efficiency of subsumption algorithms, J. ACM 32 (1985) 280–295. URL: https://doi.org/10.1145/3149.214118. doi:10.1145/3149.214118.

[16] J. D. Ullman, Database theory: Past and future, in: M. Y. Vardi (Ed.), Proceedings of the Sixth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, March 23-25, 1987, San Diego, California, USA, ACM, 1987, pp. 1–10. URL: https://doi.org/10.1145/28659.28660. doi:10.1145/28659.28660.

[17] G. Gottlob, Computing covers for embedded functional dependencies, in: M. Y. Vardi (Ed.), Proceedings of the Sixth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, March 23-25, 1987, San Diego, California, USA, ACM, 1987, pp. 58–69. URL: https://doi.org/10.1145/28659.28665. doi:10.1145/28659.28665.

[18] S. Ceri, G. Gottlob, L. Tanca, Logic Programming and Databases, Surveys in computer science, Springer, 1990. URL: https://www.worldcat.org/oclc/20595273.

[19] G. Gottlob, N. Leone, F. Scarcello, The complexity of acyclic conjunctive queries, J. ACM 48 (2001) 431–498. URL: https://doi.org/10.1145/382780.382783. doi:10.1145/382780.382783.

[20] G. Gottlob, N. Leone, F. Scarcello, Hypertree decompositions and tractable queries, J. Comput. Syst. Sci. 64 (2002) 579–627. URL: https://doi.org/10.1006/jcss.2001.1809. doi:10.1006/jcss.2001.1809.

[21] T. Eiter, G. Gottlob, H. Mannila, Disjunctive datalog, ACM Trans. Database Syst. 22 (1997) 364–418. URL: https://doi.org/10.1145/261124.261126. doi:10.1145/261124.261126.

[22] G. Gottlob, Relativized logspace and generalized quantifiers over finite ordered structures, J. Symb. Log. 62 (1997) 545–574. URL: https://doi.org/10.2307/2275546. doi:10.2307/2275546.

[23] G. Gottlob, C. Koch, Monadic datalog and the expressive power of languages for web information extraction, J. ACM 51 (2004) 74–113. URL: https://doi.org/10.1145/962446.962450. doi:10.1145/962446.962450.

[24] Panel on Past and Future of Computer Science Theory, *Christos Papadimitriou's interview,* by Giorgio Ausiello and Domenico Saccà, 2021. URL: https://drive.google.com/file/d/127ufvxvGqenCGtKZuw-woeL2Gpuk10kL/view?usp=sharing, accessed: 2021-09-09.

[25] J. van Leeuwen (Ed.), Handbook of Theoretical Computer Science, Volume A: Algorithms and Complexity, Elsevier and MIT Press, 1990.

[26] J. van Leeuwen (Ed.), Handbook of Theoretical Computer Science, Volume B: Formal

Models and Semantics, Elsevier and MIT Press, 1990. URL: https://www.sciencedirect.com/book/9780444880741/formal-models-and-semantics.
[27] C. H. Papadimitriou, Theoretical problems related to the internet, in: D. Du, P. Eades, V. Estivill-Castro, X. Lin, A. Sharma (Eds.), Computing and Combinatorics, 6th Annual International Conference, COCOON 2000, Sydney, Australia, July 26-28, 2000, Proceedings, volume 1858 of *Lecture Notes in Computer Science*, Springer, 2000, pp. 1–2. URL: https://doi.org/10.1007/3-540-44968-X_1. doi:10.1007/3-540-44968-X_1.
[28] C. H. Papadimitriou, Learning the internet, in: J. Kivinen, R. H. Sloan (Eds.), Computational Learning Theory, 15th Annual Conference on Computational Learning Theory, COLT 2002, Sydney, Australia, July 8-10, 2002, Proceedings, volume 2375 of *Lecture Notes in Computer Science*, Springer, 2002, p. 396. URL: https://doi.org/10.1007/3-540-45435-7_27. doi:10.1007/3-540-45435-7_27.
[29] E. Chastain, A. Livnat, C. H. Papadimitriou, U. V. Vazirani, Algorithms, games, and evolution, Proc. Natl. Acad. Sci. USA 111 (2014) 10620–10623. URL: https://doi.org/10.1073/pnas.1406556111. doi:10.1073/pnas.1406556111.
[30] Panel on Past and Future of Computer Science Theory, *Moshe Vardi's interview*, by Maurizio Lenzerini and Domenico Saccà, 2021. URL: https://drive.google.com/file/d/124D0IV2_H-Z3n4LGH4W1RvnbCHNnqW1Q/view?usp=sharing, accessed: 2021-09-09.
[31] C. Beeri, M. Y. Vardi, A proof procedure for data dependencies, J. ACM 31 (1984) 718–741. URL: https://doi.org/10.1145/1634.1636. doi:10.1145/1634.1636.
[32] M. Y. Vardi, The complexity of relational query languages (extended abstract), in: H. R. Lewis, B. B. Simons, W. A. Burkhard, L. H. Landweber (Eds.), Proceedings of the 14th Annual ACM Symposium on Theory of Computing, May 5-7, 1982, San Francisco, California, USA, ACM, 1982, pp. 137–146. URL: https://doi.org/10.1145/800070.802186. doi:10.1145/800070.802186.