

Управление данными: 25 лет прогнозов

С.Д. Кузнецов <kuzloc@ispras.ru>

Институт системного программирования РАН,

109004, Россия, г. Москва, ул. А. Солженицына, дом 25

Московский государственный университет имени М.В. Ломоносова,
119991 ГСП-1 Москва, Ленинские горы, МГУ имени М.В. Ломоносова, 2-й
учебный корпус, факультет ВМК

Московский физико-технический институт,

141700, Московская область, г. Долгопрудный, Институтский пер., 9

Аннотация. В октябре 2013 г. состоялась восьмая встреча исследователей в области баз данных. Первая подобная встреча прошла в феврале 1988 г., так что между ними прошло 25 лет. После каждой встречи публиковался отчет, содержащий обзор современного состояния области и программу исследований на ближайшее будущее – своего рода набор прогнозов развития исследовательской деятельности. В этой статье рассматриваются наиболее интересные прогнозы из отчетов о встречах исследования, обсуждается, насколько они оказались обоснованными, в какой мере сбылись или не сбылись. В числе рассматриваемых разнородных вопросов технологии баз данных содержатся следующие: роль специализированной аппаратуры при построении эффективных СУБД; SQL и приложения баз данных; перспективы объектно-реляционных расширений; распределенные неоднородные системы баз данных; базы данных и Web; базы и хранилища данных, OLAP и data mining; компонентная организация СУБД; критерии оптимизации запросов; самонастраиваемость и самоуправляемость СУБД; архитектура СУБД и новые аппаратные возможности: SSD, энергонезависимая память, массивно-многопоточные процессоры; специализированные СУБД; пространства данных; проблема Больших Данных и реакция на нее в сообществе баз данных; изменения в архитектуре компьютерных систем.

Ключевые слова: отчеты исследовательского сообщества баз данных; технологические прогнозы; анализ данных; машина баз данных; SQL; манифесты будущих систем баз данных; масштабируемость; неоднородность; распределенность; хранилище данных; расширяемость; оптимизация запросов; самонастраиваемость; SSD; энергонезависимая память; пространство данных; Большие Данные

DOI: 10.15514/ISPRAS-2017-29(2)-5

Для цитирования: Кузнецов С.Д. Управление данными: 25 лет прогнозов. Труды ИСП РАН, том 29, вып. 2, 2017 г., стр. 117-160. DOI: 10.15514/ISPRAS-2017-29(2)-5

1. Введение

Раз в несколько лет ведущие представители исследовательского сообщества баз данных проводят встречи, которые обычно длятся два дня. На этих встречах обсуждается и оценивается состояние дел в области баз данных и формулируются темы исследований, которые, по мнению участников, будут наиболее актуальны в ближайшие годы. По результатам встреч принято подготавливать и публиковать отчет. Такие отчеты пользуются высоким авторитетом в сообществе баз данных и оказывают серьезное влияние на развитие исследований и разработок.

К настоящему времени состоялись восемь таких встреч:

- Future Directions in DBMS Research, 1988 г., Лагуна Бич, Калифорния [1];
- NSF Invitational Workshop on the Future of Database Systems Research, Пало Альто, Калифорния, 1990 [3];
- NSF Workshop on the Future of Database Systems Research, Пало Альто, Калифорния, 1995 [4];
- Workshop on Strategic Directions in Computing Research, Кембридж, шт. Массачусетс, 1996 [5];
- Asilomar Meeting, Асиломар, неподалеку от г. Монтерей, Калифорния, 1998 [6];
- Lowell Meeting, Лоуэлл, шт. Массачусетс, 2003 [7];
- Claremont Meeting, Беркли, Калифорния, отель Claremont Resort, 2008 [9];
- Beckman Meeting, Ирвин, Калифорния, Бекманский университетский центр, 2013 [10].

В табл. 1 собрана информация о времени и месте проведения этих встреч, их участниках, официальных и неофициальных публикациях отчетов.

Табл. 1. Сводная информация о прошедших встречах

Table 1. Summary information on meetings of database research community

Время и место встречи	Список участников на английском языке	Список участников на русском языке	Официальная публикация	Публикация на русском языке
1. 4-5 февраля 1988 г. Лагуна Бич, шт. Калифорния «Laguna Beach meeting»	1. Philip A. Bemstein 2. Umeshwar Dayal 3. David J. DeWitt 4. Dieter Gawlick 5. Jim Gray 6. Matthias Jarke 7. Bruce G. Lindsay 8. Peter C. Lockemann 9. David Maier 10. Erich J. Neuhold 11. Andreas Reuter 12. Lawrence A. Rowe 13. Hans J. Schek 14. Joachim W. Schmidt	1. Филипп Бернштейн 2. Умешвар Дайал 3. Дэвид Девитт 4. Дитер Гавлик 5. Джим Грей 6. Маттиас Ярке 7. Брюс Линдсей 8. Питер Локман 9. Дэвид Майер 10. Эрик Ньюхолд 11. Андреас Рейтер 12. Лоуренс Роув 13. Ханс-Йорг Шек 14. Йохим Шмидт	Future Directions in DBMS Research - The Laguna Beach Participants. ACM SIGMOD Record, 18(1):17-26, 1989	Сергей Кузнецов. Будущие направления исследований в области баз данных: десять лет спустя, 1999, http://citforum.ru/database/articles/future_01.shtml

	15. Michael Schrefl 16. Michael Stonebraker	15. Михаэль Шрефл 16. Майкл Стоунбрейкер		
2. 22-24 февраля 1990 г. Пало Альто, шт. Калифорния. NSF Invitational Workshop on the Future of Database System Research «Lagunita meeting 1»	1. Michael Brodie 2. Peter Buneman 3. Mike Carey 4. Ashok Chandra 5. Hector Garcia-Molina 6. Jim Gray 7. Ron Fagin 8. Dave Lomet 9. Dave Maier 10. Marie Ann Niemat 11. Avi Silberschatz, 12. Michael Stonebraker 13. Irv Traiger 14. Jeff Ullman 15. Gio Wiederhold 16. Carlo Zaniolo 17. Maria Zemankova.	1. Майкл Броуди 2. Питер Бьюнман 3. Майк Кэри 4. Ашок Чандра 5. Гектор Гарсиа-Молина 6. Джим Грей 7. Рон Фейджин 8. Дейв Ломе 9. Дейв Майер 10. Мэри Энн Наймэт 11. Эви Зильбершац 12. Майкл Стоунбрейкер 13. Ирв Трайгер 14. Джефф Ульман 15. Джо Видерхолд 16. Карло Заниоло 17. Мария Земанкова	Avi Silberschatz, Michael Stonebraker, Jeff Ullman, editors. Database Systems: Achievements and Opportunities Comm. of the ACM, 34(10):110-120, 1991	
3. 26-27 мая 1995 г. Пало Альто, шт. Калифорния. NSF Workshop on the Future of Database Systems Research «Lagunita meeting 2»	1. Phil Bernstein 2. Ron Brachman 3. Mike Carey 4. Rick Cattell 5. Hector Garcia-Molina 6. Laura Haas 7. Dave Maier 8. Jeff Naughton 9. Michael Schwartz 10. Pat Selinger 11. Avi Silberschatz 12. Mike Stonebraker 13. Jeff Ullman 14. Patrick Valduriez 15. Moshe Vardi 16. Jennifer Widom 17. Gio Wiederhold 18. Marianne Winslett 19. Maria Zemankova	1. Филипп Бернштейн 2. Рон Брахман 3. Майкл Кэри 4. Рик Каттел 5. Гектор Гарсиа-Молина 6. Лаура Хаас 7. Дейв Майер 8. Джефф Нотон 9. Майкл Шварц 10. Пат Селинджер 11. Эви Зильбершац 12. Майкл Стоунбрейкер 13. Джефф Ульман 14. Патрик Вальдурец 15. Мойше Варди 16. Дженифер Вайдом 17. Джо Видерхолд 18. Марианна Винслетт 19. Мария Земанкова	Avi Silberschatz, Mike Stonebraker, Jeff Ullman. Database Research: Achievements and Opportunities into the 21st Century. ACM SIGMOD Record, 25(1):52-63, 1996	Эви Зильбершац, Майкл Стоунбрейкер, Джефф Ульман. Базы данных: достижения и перспективы на пороге 21-го столетия. Новая редакция: Сергей Кузнецов, 2009 г., http://citforum.ru/database/classics/nfs_report/
4. 14-15 июня 1996 г. MIT, Кембридж, шт. Массачусетс ACM Workshop on Strategic Directions in Computing Research «Cambridge meeting»	1. Jose Blakeley 2. Peter Buneman 3. Umesh Dayal 4. Tomasz Imielinski 5. Sushil Jajodia 6. Hank Korth 7. Guy Lohman, 8. Dave Lomet 9. Dave Maier 10. Frank Manola 11. Tamer Ozsu 12. Raghu Ramakrishnan 13. Krithi Ramamritham 14. Hans Schek 15. Avi Silberschatz 16. Rick Snodgrass 17. Jeff Ullman 18. Jennifer Widom, 19. Stan Zdonik	1. Хосе.Блейкли 2. Питер Бьюнман 3. Умеш.Дайал 4. Томаш.Имилинский 5. Сушил.Джаджодиа 6. Хэнк.Корт 7. Гай.Лохман 8. Дейв.Ломе 9. Дейв Майер 10. Френк.Манола 11. Тамер.Озсу 12. Раджу Рамакришнан 13. Крити.Рамамритан 14. Ганс.Шек 15. Эви.Зильбершац 16. Рик Снодграсс 17. Джефф Ульман 18. Дженифер.Вайдом 19. Стенли.Здоник	Avi Silberschatz, Stan Zdonik, et al. «Strategic Directions in Database Systems: Breaking Out of the Box». ACM Computing Surveys, 28(4):764-778, 1996	А.Зильбершац, С.Здоник и др. Стратегические направления в системах баз данных Перевод: М.Р. Коголовский Новая редакция: Сергей Кузнецов, 2009 г., http://citforum.ru/database/classics/nfs_report2/
5. 19-21 августа 1998 г.	1. Phil Bernstein 2. Michael Brodie 3. Stefano Ceri	1. Филипп Бернштейн 2. Майкл Броуди 3. Стефано Чери	Phil Bernstein, Michael Brodie et al. The	Филипп Бернштейн, Майкл Броуди и

Асиломар, г. Пасифик Гров, шт. Калифорния «Asilomar meetong»	4. David DeWitt 5. Mike Franklin 6. Hector Garcia-Molina 7. Jim Gray 8. Jerry Held 9. Joe Hellerstein 10. H. V. Jagadish 11. Michael Lesk 12. Dave Maier 13. Jeff Naughton 14. Hamid Pirahesh 15. Mike Stonebraker 16. Jeff Ullman	4. Дэвид Девитт 5. Майк Франклин 6. Гектор Гарсиа-Молина 7. Джим Грей 8. Джерри Хелд 9. Джо Хеллерштейн 10. Х.В. Ягадиш 11. Майкл Леск 12. Дейв Майер 13. Джефф Нотон 14. Хамид Пирамеш 15. Майк Стоунбрейкер 16. Джефф Ульман	Asilomar Report on Database Research. ACM SIGMOD Record, 27(4):74-80, 1998	др. Асиломарский отчет об исследованиях в области баз данных. Перевод: Сергей Кузнецов, 1999, http://citforum.ru/database/digest/asil_01.shtml
6. 4-6 мая 2003 г. Лоуэлл, шт. Массачусетс «Lowell meeting»	1. Serge Abiteboul 2. Rakesh Agrawal 3. Phil Bernstein 4. Mike Carey 5. Stefano Ceri 6. Bruce Croft 7. David DeWitt 8. Mike Franklin 9. Hector Garcia Molina 10. Dieter Gawlick 11. Jim Gray 12. Laura Haas 13. Alon Halevy 14. Joe Hellerstein 15. Yannis Ioannidis 16. Martin Kersten 17. Michael Pazzani 18. Mike Lesk 19. David Maier 20. Jeff Naughton 21. Hans Schek 22. Timos Sellis 23. Avi Silberschatz 24. Mike Stonebraker 25. Rick Snodgrass 26. Jeff Ullman 27. Gerhard Weikum 28. Jennifer Widom 29. Stan Zdonik	1. Серж Абитебуль 2. Ракеш Агравал 3. Филипп Бернштейн 4. Майк Кэри 5. Стефано Чери 6. Брюс Крофт 7. Дэвид Девитт 8. Майк Франклин 9. Гектор Гарсиа-Молина 10. Дитер Гавлик 11. Джим Грей 12. Лаура Хаас 13. Элон Хэлеви 14. Джо Хеллерштейн 15. Янис Ионнидис 16. Мартин Керстен 17. Майкл Паззани 18. Майк Леск 19. Дэвид Мейер 20. Джефф Нотон 21. Ганс Шек 22. Тимос Селлис 23. Эви Зильбершац 24. Майкл Стоунбрейкер 25. Рик Снодграсс 26. Джефф Ульман 27. Герхард Вейкум 28. Дженифер Вайдом 29. Стен Здоник	Serge Abiteboul, Rakesh Agrawal et al. The Lowell Database Research Self-Assessment. Communication s of the ACM, 48(5):111-118, 2005	Сергей Кузнецов. Крупные проблемы и текущие задачи исследований в области баз данных, 2005, http://citforum.ru/database/articles/problems/
7. 29-30 мая 2008 г. Беркли, шт. Калифорния, Клермонт Ресорт «Claremont meeting»	1. Rakesh Agrawal 2. Anastasia Ailamaki 3. Philip A. Bernstein 4. Eric A. Brewer 5. Michael J. Carey 6. Surajit Chaudhuri 7. AnHai Doan 8. Daniela Florescu 9. Michael J. Franklin 10. Hector Garcia Molina 11. Johannes Gehrke 12. Le Gruenwald 13. Laura M. Haas 14. Alon Y. Halevy 15. Joseph Hellerstein 16. Yannis E. Ioannidis 17. Hank F. Korth 18. Donald Kossmann 19. Samuel Madden 20. Roger Magoulas	1. Ракеш Агравал 2. Анастасия Айламаки 3. Филипп Бернштейн 4. Эрик Брюер 5. Майкл Кэри 6. Сураджит Чаудхари 7. Анхай Доан 8. Даниела Флореску 9. Майкл Франклин 10. Гектор Гарсиа-Молина 11. Иоханнес Герке 12. Ле Грюнвальд 13. Лаура Хаас 14. Элон Хэлеви 15. Джозеф Хеллерштейн 16. Янис Ионнидис 17. Хэнк Корт 18. Дональд Коссмман 19. Самуэль Мэдден 20. Роджер Магулас	Rakesh Agrawal, Anastasia Ailamaki,et al. The Claremont Report on Database Research. Communication s of the ACM, 52(6):56-65, 2009.	Ракеш Агравал, Анастасия Айламаки др. Клермонтский отчет об исследованиях в области баз данных. Пересказ и комментарии: Сергей Кузнецов, 2008, http://citforum.ru/database/articles/claremont_report/

	21. Beng Chin Ooi 22. Tim O'Reilly 23. Raghuram Ramakrishnan 24. Sunita Sarawagi 25. Michael Stonebraker 26. Alexander S. Szalay 27. Gerhard Weikum	21. Бен Чин Ой 22. Тим О'Рейли 23. Раджу Рамакришнан 24. Сунита Сарагави 25. Майкл Стоунбрейкер 26. Александр Шалай 27. Герхард Вейкум		
8. 14-15 октября 2013 г. Ирвин, шт. Калифорния, Бекманский центр университета в Ирвине «Beckman meeting»	1. Daniel Abadi 2. Rakesh Agrawal 3. Anastasia Ailamaki 4. Magdalena Balazinska 5. Philip A. Bernstein 6. Michael J. Carey 7. Surajit Chaudhuri 8. Jeffrey Dean 9. AnHai Doan 10. Michael J. Franklin 11. Johannes Gehrke 12. Laura M. Haas 13. Alon Y. Halevy 14. Joseph Hellerstein 15. Yannis E. Ioannidis 16. H.V. Jagadish 17. Donald Kossmann 18. Samuel Madden 19. Sharad Mehrotra 20. Tova Milo 21. Jeffrey F. Naughton 22. Raghuram Ramakrishnan 23. Volker Markl 24. Christopher Olston 25. Beng Chin Ooi 26. Christopher Re 27. Dan Suciu 28. Michael Stonebraker 29. Todd Walter 30. Jennifer Widom	1. Дэниел Абади 2. Ракеш Агравал 3. Анастасия Айламаки 4. Магдалена Балазинска 5. Филип А. Берштейн 6. Майкл Дж. Кэри 7. Сурадждит Чаудхари 8. Джеффри Дин 9. Анхай Доан 10. Майкл Дж. Франклин 11. Йоханнес Герке 12. Лаура М. Хаас 13. Элон И. Хэлеви 14. Джозеф Хеллерштейн 15. Янис Е. Ионнидис 16. Х.В. Ягадиш 17. Дональд Коссманн 18. Самуэль Мэдден 19. Шарад Мехротра 20. Тоба Мило 21. Джеффри Нотон 22. Раджу Рамакришнан 23. Волкер Маркл 24. Кристофер Олстон 25. Бен Чин Ой 26. Кристофер Ре 27. Дан Сучиу 28. Майкл Стоунбрейкер 29. Тодд Валтер 30. Дженифер Вайдом	Daniel Abadi, Rakesh Agrawal et al. The Beckman Report on Database Research. SIGMOD Record, September, 43(3):61-70, 2014	Дэниел Абади, Ракеш Агравал и др. Бекманский отчет об исследованиях в области баз данных. Перевод: Сергей Кузнецов, 2017

В разные годы во встречах принимали участие разные специалисты, но некоторые известные исследователи внесли особенно заметный вклад в эту работу. В табл. 2 перечислены исследователи, принявшие участие в двух и большем числе встреч.

Табл. 2. Наиболее активные участники встреч
Table 2. The most active participants

Число встреч	Имя участника	В каких встречах участвовал
7	Майкл Стоунбрейкер	1, 2, 3, 5, 6, 7, 8
6	Дэвид Майер	1, 2, 3, 4, 5, 6
	Филип Берштейн	1, 3, 5, 6, 7, 8
5	Гектор Гарсиа-Молина	2, 3, 5, 6, 7
	Джеффри Ульман	2, 3, 4, 5, 6,
	Майкл Кэри	2, 3, 6, 7, 8
4	Эви Зильбершцац	2, 3, 4, 6
	Джеффри Нотон	3, 5, 6, 8

	Дженифер Вайдом	3, 4, 6, 8
	Джим Грей (пропал без вести в 2007 г.)	1, 2, 5, 6
	Джо Хеллерштейн	5, 6, 7, 8
	Лаура Хаас	3, 6, 7, 8
	Майкл Франклин	5, 6, 7, 8
3	Дэвид Девиитт	1, 5, 6
	Ганс Шек	1, 4, 6
	Раджу Рамакришнан	4, 7, 8
	Ракеш Агравал	6, 7, 8
	Янис Ионнидис	6, 7, 8
2	Элон Хэлеви	7, 8
	Анастасия Айламаки	7, 8
	Анхай Доан	7, 8
	Бен Чин Ой	7, 8
	Дэвид Ломе	2, 4
	Дитер Гавлик	1, 6
	Дональд Коссманн	7, 8
	Герхард Вейкум	6, 7
	Джо Видерхолд	2, 3
	Х.В. Ягадиш	5, 6
	Хэнк Корт	4, 7
	Йоханнес Герке	7, 8
	Мария Земанкова	2, 3
	Майкл Броуди	2, 5
	Питер Бьюнман	2, 4
	Рик Снодграсс	4, 6
	Сэмюэль Мэдден	7, 8
Стенли Здоник	4, 6	
Стефано Чери	5, 6	
Сурадждит Чаудхари	7, 8	
Умешвар Дайал	1, 4	

Легко видеть, что между первой и последней встречами прошло 25 лет, так что Бекманская встреча была юбилейной. В честь этого юбилея с осознанием важности для мирового исследовательского сообщества регулярных отчетов о перспективных исследованиях в области баз данных я решил написать эту статью. В некотором смысле статья должна предоставить читателям общую картину развития технологии баз данных за четверть века.

В статье обсуждаются самые интересные, с точки зрения автора, прогнозы, содержащиеся в отчетах о встречах прошлых лет [1–9], насколько они были реалистичными, точными, деловыми и прагматичными или, наоборот, утопичными, конъюнктурными или маркетинговыми, какие прогнозы сбылись, а какие канули в вечность и т.д. Почти не затрагивается отчет [10] о последней по счету встрече, состоявшейся осенью 2013 г. в Калифорнии, оценивать точность прогнозов которой еще не пришло время.

2. Встреча в Лагуна Бич: 1988

Первая встреча состоялась 4-5 февраля 1988 г. в г. Лагуна Бич, шт. Калифорния. Вот фрагмент табл. 1, описывающей детали этой встречи.

Время и место встречи	Список участников на английском языке	Список участников на русском языке	Официальная публикация	Публикация на русском языке
4-5 февраля 1988 г. Лагуна Бич, шт. Калифорния «Laguna Beach meeting»	17. Philip A. Bemstein 18. Umeshwar Dayal 19. David J. DeWitt 20. Dieter Gawlick 21. Jim Gray 22. Matthias Jarke 23. Bruce G. Lindsay 24. Peter C. Lockemann 25. David Maier 26. Erich J. Neuhold 27. Andreas Reuter 28. Lawrence A. Rowe 29. Hans J. Schek 30. Joachim W. Schmidt 31. Michael Schrefl 32. Michael Stonebraker	17. Филипп Бернштейн 18. Умешвар Дайал 19. Дэвид Девитт 20. Дитер Гавлик 21. Джим Грей 22. Маттиас Ярке 23. Брюс Линдсей 24. Питер Локман 25. Дэвид Майер 26. Эрик Ньюхолд 27. Андреас Рейтер 28. Лоуренс Роув 29. Ханс-Йорг Шек 30. Йохим Шмидт 31. Михаэль Шрефл 32. Майкл Стоунбрейкер	Future Directions in DBMS Research - The Laguna Beach Participants. ACM SIGMOD Record, 18(1):17-26, 1989	Сергей Кузнецов. Будущие направления исследований в области баз данных: десять лет спустя, 1999, http://citforum.ru/database/articles/future_01.shtml

Отчет о результатах встречи официально опубликован в 1989 г. [1] (официальные публикации отчетов о встречах, которым посвящена данная статья, обычно появляются позже неофициальных публикаций в виде технических отчетов различных организаций; в частности, исходный общедоступный вариант отчета Лагуна Бич находится на сайте университета Беркли: <http://www.icsi.berkeley.edu/pubs/techreports/tr-88-001.pdf>, дата доступа 24.03.2017). На русский язык отчет Лагуна Бич полностью не переводился. Подробный пересказ отчета с комментариями можно найти в [2].

Далее в этом разделе будут кратко рассмотрены наиболее интересные (с точки зрения автора) прогнозы, содержащиеся в [1], и то, насколько они сбылись к теперешнему времени или насколько актуальными они остаются сейчас.

2.1. Распространение аналитики

В отчете предсказывалось «широкое внедрение систем поддержки принятия решений по мере снижения цен аппаратуры». Подход к построению «систем поддержки принятия решений» (decision support system, DSS) возник в 70-е гг. прошлого века на основе работ, выполнявшихся еще в 60-е гг. [11]. Термин DSS никогда не имел однозначного и четкого определения, как не имеет его и в настоящее время. Однако уже в конце 80-х гг. специалистам в области баз данных было понятно, что очень во многих случаях принятие решений в различных организациях должно основываться на анализе данных.

В 90-е гг. XX века были развиты теоретические основы и технология хранилищ данных (datawarehouse) и оперативной аналитической обработки данных (online analytical data processing, OLAP). В первом десятилетии XXI века появились горизонтально масштабируемые массивно-параллельные аналитические СУБД, возникла технология map/reduce (см., например, [12]). Тогда же активно развивались (и продолжают развиваться в настоящее время) методы интеллектуального анализа данных (data mining, text mining и т.д.). К анализу активно привлекаются данные, генерируемые различными пользователями Internet, в частности, социальных сетей.

Даже если не затрагивать варианты DSS, напрямую не основанные на анализе данных (например, разного рода экспертные системы), то можно сказать, что прогноз о широком внедрении DSS полностью оправдался. Использование аналитики с целью поддержки принятия решений сегодня повсеместно.

2.2. Малая перспективность специализированной аппаратуры

В отчете отмечалось, что «в области управления базами данных компьютеры общего назначения более перспективны, чем специализированная аппаратура». Заметим, что к 1988 г. стало ясно, что японский проект компьютеров пятого поколения [13] не то чтобы с треском, но таки провалился (по крайней мере, в связи с разработкой специализированной аппаратуры для построения особенно эффективных и мощных СУБД).

У японцев предполагалась разработка двух видов специализированных аппаратных средств для поддержки СУБД: процессоры, в которых операции реляционной алгебры поддерживались на уровне системы команд, и магнитные диски с фиксированными головками, в головки которых должны были встраиваться микропроцессоры для фильтрации данных «на лету» по мере их чтения с дисков («умные» диски).

Первый подход не удался по разным причинам (скорее всего, можно существенно расширить приводимый список):

- поскольку базы данных сохраняются во внешней дисковой памяти, используемые структуры данных оптимизированы в расчете на это; для непосредственного применения команд процессора недостаточно переместить данные с диска в основную память;
- таблицы SQL-ориентированных баз данных (а к началу проекта пятого поколения уже было понятно, что будущее за SQL-ориентированными СУБД) часто настолько велики, что не могут целиком поместиться в основной памяти компьютера; итерационное применение команд «реляционной алгебры» далеко не всегда возможно и/или эффективно;
- наличие алгоритмов выполнения реляционных операций, «зашитых» в машинные команды, противоречит идее гибкой оптимизации запросов,

когда оптимизатор выбирает наиболее подходящий алгоритм в зависимости от текущего состояния базы данных;

- при сохранении баз данных на магнитных дисках основное время выполнения запроса уходит на выполнение обменов с внешней памятью; экономия на времени обработки данных в основной памяти часто бывает несущественной.

Что касается второго подхода, то он не нашел широкого применения, прежде всего, по экономическим причинам. Легко видеть, что объем необходимой аппаратуры в дисковом устройстве с фиксированными головками во много раз больше, чем в традиционных жестких дисках. Поэтому и стоимость таких устройств значительно выше (а надежность – ниже). Кроме того, по разным причинам (в частности, из-за проблем распределения основной памяти) в традиционной архитектуре SQL-ориентированных СУБД предполагается блочный обмен данными с внешним хранилищем баз данных, и поэтому не очень понятно, как СУБД может получить от «умного» диска отфильтрованные на лету данные.

Интересно, что в известной статье 1992 г. [14] Девиит и Грей авторитетно указывали на провал подхода «умных» дисков и предрекали, что будущие высокопроизводительные параллельные СУБД будут по-прежнему опираться на использование дисков с подвижными головками. Но уже 10 лет спустя в своем интервью [15] Джим Грей говорил, что из-за стремительного удешевления аппаратных средств (в частности, мощных микропроцессоров) теперь возможен перенос СУБД целиком на головку магнитного диска. Как видно, эта идея распространения не нашла (думаю, что Джим фантазировал, потому что такая архитектура даже на первый взгляд кажется очень проблематичной).

И вообще, список решений XXI века для управления базами данных, основанных на использовании специализированной аппаратуры, выглядит очень скромно.

Database appliance (программно-аппаратные комплексы для управления базами данных). По определению Gartner [16], database appliance – это заранее собранный и/или сконфигурированный набор оборудования (серверы, память, накопители и каналы ввода/вывода), программное обеспечение (операционная система, СУБД и т.д.), средства обслуживания и поддержки. Определение достаточно расплывчато; в соответствии с ним, database appliance является скорее маркетинговым, а не технологическим понятием, и это, на мой взгляд, отражает сложившуюся действительность. В некоторых продуктах этой категории (наиболее ярким примером является Oracle Exadata Database Machine [17]) используются специализированные программно-аппаратные компоненты, но сами продукты являются универсальными, равно пригодными для обработки и транзакционных, и аналитических рабочих нагрузок. Другой подход исповедует Майкл Стоубрейкер [18]. По его мнению, в состав database appliance не должна входить специализированная аппаратура, но зато

специализированным должно быть программное обеспечение самой СУБД. В любом случае, database appliance никак не похожи на машины баз данных 1980-х.

Использование графических процессоров. Как известно, в настоящее время очень модно (и, наверное, в ряде случаев перспективно) пытаться использовать GPU в областях, отличных от обработки графики. В частности, растет популярность применения GPU для обработки запросов в СУБД, ориентированных на хранение данных в основной памяти (например, [19]). По ряду причин, у меня нет уверенности в перспективности этого подхода (хотя бы потому, что разные графические процессоры слишком различаются, а разработчикам СУБД часто для получения эффективности требуется использовать уникальные возможности GPU). Кроме того, здесь мы имеем дело совсем не со специализированной аппаратурой, разработанной именно для поддержки СУБД.

Можно считать, что прогноз оказался достаточно точным.

2.3 Широкое распространение ОС с микроядерной архитектурой

Действительно, в 1980-е гг. многим (включая меня) казалось, что *будущее за микроядерными операционными системами с более развитыми и современными функциональными возможностями*, чем у тогдашнего (да и теперешнего) фаворита – семейства ОС UNIX. Выполнялись многочисленные проекты, среди которых мне ближе других Chorus [20], CLOS [21] и Mach [22]. По разным причинам большая часть этих проектов не удалась (по крайней мере, те результаты, на которые надеялись разработчики, получить не удалось).

Фактически, обе довлеющие сегодня операционные системы Linux и Windows микроядерными не являются. Как и 20 лет тому назад, единственной коммерческой микроядерной системой является QNX [23], но эта ОС (семейство ОС) является нишевой и не может служить подтверждением «широкого» распространения микроядерного подхода.

Интересную работу в XXI веке провел Эндрю Таненбаум – проект Minix 3 [24]. Это современный, хорошо проработанный проект, который вряд ли ждет широкое практическое использование (хотя бы потому, что в нем реализуется только API Posix). Занятно, что основная разработка Minix 3 завершилась в 2014 г. одновременно с уходом Таненбаума на пенсию.

Так что этот прогноз участников встречи в Лагуна Бич не оправдался.

2.4 Ужасный интерфейс SQL

Участники встречи отмечали, что было бы хорошо *улучшить существующий ужасный интерфейс SQL для встраиваемых и динамических запросов*. Заметим, что это происходило за четыре года до принятия стандарта SQL-92 (SQL:1992), в котором соответствующие возможности были полностью и

окончательно определены. С ранней историей стандартов языка SQL можно ознакомиться, например, в [25].

Идеи встраивания операторов языка SQL в языки программирования и динамической компиляции SQL появились еще на заре SQL в 1970-е гг. во время выполнения в IBM проекта System R (история этого замечательного проекта лучше всего описана в [26]). Причины же появления средств встраивания и динамической компиляции в SQL времени System R кратко и четко описаны в интервью одного из основных разработчиков начального SQL Дона Чемберлина [27].

Конечно, средства встраиваемого и динамического SQL трудно назвать красивыми и/или элегантными (впрочем, с моей точки зрения, то же относится и к языку SQL в целом). Но за время их существования (более 40 лет) ничего принципиально нового придумать не удалось. Широко распространено использование интерфейсов уровня вызова функций ODBC и JDBC. Но эти интерфейсы не избавляют от необходимости сохранять в программе или динамически формировать строки, содержащие тексты требуемых операторов SQL, а именно наличие в программе таких строк делает столь «ужасными» традиционные средства SQL для поддержки встраиваемых и динамически формируемых операторов SQL.

Так что улучшить интерфейс SQL так и не удалось.

3. Первая встреча в Пало Альто: 1990

Вторая встреча исследователей в области баз данных состоялась в Стэнфордском университете (Пало Альто, шт. Калифорния) 22-24 февраля 1990 г. в рамках организованного Национальным научным фондом США симпозиума «Future of Database System Research». Эта и следующая встречи неформально называются «встречами Лагунита», поскольку проходили вблизи одноименного искусственного озера. Вот соответствующий фрагмент табл. 1.

Время и место встречи	Список участников на английском языке	Список участников на русском языке	Официальная публикация
22-24 февраля 1990 г. Пало Альто, шт. Калифорния. NSF Invitational Workshop on the Future of Database System Research «Lagunita meeting 1»	1. Michael Brodie 2. Peter Buneman 3. Mike Carey 4. Ashok Chandra 5. Hector Garcia-Molina 6. Jim Gray 7. Ron Fagin 8. Dave Lomet 9. Dave Maier 10. Marie Ann Niemat 11. Avi Silberschatz, 12. Michael Stonebraker 13. Irv Traiger 14. Jeff Ullman 15. Gio Wiederhold 16. Carlo Zaniolo 17. Maria Zemankova.	1. Майкл Броуди 2. Питер Бьюнман 3. Майк Кэри 4. Ашок Чандра 5. Гектор Гарсиа-Молина 6. Джим Грей 7. Рон Фейджин 8. Дейв Ломе 9. Дейв Майер 10. Мэри Энн Наймэт 11. Эви Зильбершац 12. Майкл Стоунбрейкер 13. Ирв Трайджер 14. Джеф Уллман 15. Джо Видерхолд 16. Карло Заниоло 17. Мария Земанкова	Avi Silberschatz, Michael Stonebraker, Jeff Ullman, editors. Database Systems: Achievements and Opportunities. Communications of the ACM, 34(10):110-120, 1991

Официально отчет опубликован в 1991 г. [3], на русский язык не переводился и не пересказывался. Предварительная неформальная и общедоступная (если не бояться postscript) публикация размещена на infolab.stanford.edu/~hector/lagi.ps, дата доступа 24.03.2017.

Вот самые интересные прогнозы из Lagunita Report.

3.1 Будущие приложения баз данных

В отчете предсказывалось, что *приложения баз данных следующего поколения будут иметь мало общего с сегодняшними приложениями баз данных для обработки бизнес-данных. Они будут затрагивать гораздо больше данных; потребуют поддержки расширений системы типов данных, мультимедийных данных, сложных объектов, обработки правил и архивной системы хранения; в СУБД потребуется переосмысление алгоритмов выполнения большинства операций.*

Для правильной оценки этого прогноза необходимо восстановить общий контекст, в котором проходила первая встреча в Пало Альто. Только что был опубликован Манифест систем объектно-ориентированных баз данных (Первый манифест) [28], авторы которого, по сути, отрицали дальнейшую перспективность технологий реляционных и SQL-ориентированных баз данных, упрекая их в ограниченности и неспособности удовлетворять требования будущих приложений. Готовился ответный Манифест систем баз данных третьего поколения (Второй манифест) [29], в котором не отрицались конструктивные аспекты [28] (в частности, потребность в наличии у будущих СУБД развитой и расширяемой системы типов данных), но утверждалось, что требуемые возможности с СУБД можно получить путем эволюции имеющейся технологии баз данных. Кроме того, у Майкла Стоунбрейкера имелась коммерческая СУБД Illustra, основанная на исследовательском прототипе Postgres, в которой уже поддерживалась большая часть возможностей, упомянутых в прогнозе (история Postgres-Illustra хорошо описана Стоунбрейкером в [30]).

На встрече присутствовали представители обоих лагерей. Поэтому вполне естественно появление в отчете прогноза в обтекаемой формулировке, отражающей общие взгляды исследователей из реляционного и объектно-ориентированного миров и не выпячивающей их принципиальные противоречия. Несмотря на обтекаемость формулировки прогноза, в ней заметно влияние Стоунбрейкера, который с должным основанием всегда имел авторитет в сообществе баз данных.

К началу XXI века после драматических событий 1990-х, когда ведущие вендоры SQL-ориентированных СУБД приняли концепцию объектно-реляционных баз данных, и был выпущен стандарт SQL:1999, в котором эта концепция была затверждена, технология объектно-ориентированных СУБД стала стремительно терять популярность. Сегодня, несмотря на неоднократные

попытки возродить притягательность ООСУБД для широкой аудитории разработчиков баз данных и их приложений, эти системы являются нишевыми. С другой стороны, в современных ведущих SQL-ориентированных СУБД реализованы практически все возможности, отсутствие которых мотивировало авторов [28]. Можно сказать, что Второй манифест победил. Можно ли поэтому сказать, что прогноз, обсуждаемый в этом подразделе, полностью сбился?

Безусловно, сегодняшние приложения баз данных в среднем работают с гораздо большими объемами данных, чем в начале 1990-х. Разработчики баз данных и их приложений имеют возможности определять собственные типы данных и произвольно сложной внутренней структурой и поведением. Но похоже, что сегодня, как и 10 лет тому назад [31], пользователи не слишком стремятся определять собственные типы данных внутри баз данных. Скорее, можно говорить о том, что определяемые пользователями типы данных применяют сами вендоры SQL-ориентированных СУБД для расширения их функциональных возможностей. Так что оценка потребности разработчиков баз данных и их приложений в расширяемой системе типов и сложных объектах оказалась преувеличенной.

Сложная система правил осталась прерогативой Postgres (теперь еще и PostgreSQL), а в подавляющем большинстве современных СУБД правила используются в объеме определенного в стандарте SQL механизма триггеров. Поддержка мультимедийных данных реализуется вендорами СУБД за счет наличия стандартного типа BLOB и дополнительных типов данных, реализуемых с помощью стандартного механизма определения типов данных. Наличие архивной системы хранения предполагалось в Postgres, но в современных SQL-ориентированных СУБД такая система явным образом отсутствует. Наконец, мне неизвестно, чтобы какой-либо производитель SQL-ориентированных СУБД коренным образом изменил алгоритмы выполнения операций.

Можно считать, что прогноз сбился частично.

3.2 Неоднородность, распределенность, масштабируемость

Участники встречи полагали, что для кооперации различных организаций при решении научных, технических и коммерческих проблем потребуются масштабируемые, неоднородные, распределенные базы данных. Сложными проблемами технологии распределенных СУБД являются несогласованность баз данных, безопасность и масштабируемость.

Тематике интеграции неоднородных баз данных, начиная с 1980-х гг., посвящалось множество исследований, результаты которых описывались во множестве публикаций. Здесь речь идет о так называемой виртуальной интеграции разнородных баз данных. В отличие от физической интеграции нескольких баз данных с целью построения хранилища данных, основанной на тщательно разработанных процедурах ETL (Extract, Transform, Load), при виртуальной интеграции дополнительное хранилище данных не создается,

данные извлекаются из разных источников и интегрируются «на лету» при обработке интеграционным сервером запросов к «интегрированной» базе данных. Понятно, что интегрируемые источники данных в общем случае распределены в локальной или глобальной сети, могут поддерживаться различными СУБД и даже основываться на разных моделях данных.

Многолетние исследования привели к разработке многочисленных подходов и методов виртуальной интеграции баз данных (не очень старый и краткий обзор см. в [32]), но ни одна из работ не привела к созданию хотя бы относительно безупречной технологии. Несмотря на наличие на рынке многих интеграционных продуктов, широким спросом они не пользуются, и для сообщества исследователей в области данных тематика интеграции разнородных данных сегодня не является магистральным направлением, хотя потребность в виртуальной интеграции гетерогенных источников данных периодически возникает в разных областях человеческой деятельности. Так что нельзя сказать, что прогноз сбился, но и нельзя считать его неоправданным.

Вместе с тем, в XXI веке технология распределенных систем смыкается с технологией баз данных в нескольких направлениях, отличных от интеграции данных. Первое направление – специализированные массивно-параллельные SQL-ориентированные СУБД. В первом десятилетии XXI века были разработаны исследовательские прототипы аналитической массивно-параллельной СУБД C-Store с хранением таблиц по столбцам [33] и транзакционной массивно-параллельной СУБД H-Store с хранением данных в основной памяти [34]. Обе эти системы основывались на подходе shared-nothing (без использования общих ресурсов) с целью обеспечения горизонтальной масштабируемости при возрастании числа узлов в системе, обе показали хорошие результаты на соответствующих стандартных тестовых наборах.

Оба проекта были коммерциализованы. На основе результатов проекта C-Store была основана компания Vertica, поглощенная в 2011 г. Hewlett-Packard [35]. Результаты проекта H-Store используются в коммерческом продукте VoltDB, разрабатываемом и поддерживаемом одноименной компанией [36].

Имеется ряд других SQL-ориентированных СУБД, изначально предназначенных для использования в массивно-параллельной среде. Конечно, под массивно-параллельной средой здесь понимаются кластеры, но что такое кластер, как не однородная специализированная локальная сеть? Поэтому в массивно-параллельных архитектурах СУБД активно используются методы распределенных систем: репликация данных, распределенные транзакции и т.д. Другое направление – глобально распределенные СУБД категории NoSQL. Это направление бурно развивается и непрерывно изменяется, поэтому трудно охарактеризовать его текущее состояние. Тем не менее, думаю, что ряд характеристик, указанных в [37], остается справедливым. Наверное, наиболее общими чертами этого направления является стремление к обеспечению горизонтальной масштабируемости систем и высокого уровня доступности данных за счет репликации. Доступность обеспечивается за счет отказа строгой транзакционности (в глобально распределенной среде строгая согласованность

реплик стоит слишком дорого). Несмотря на общее название направления, в системах все чаще поддерживаются ограниченные диалекты SQL.

4. Вторая встреча в Пало Альто: 1995

Третья по счету встреча исследователей в области баз данных также состоялась в Пало Альто в рамках второго спонсированного Национальным научным фондом США симпозиума Future of Database Systems Research (Лагунита 2). Подробности в приводимом фрагменте табл. 1.

Время и место встречи	Список участников на английском языке	Список участников на русском языке	Официальная публикация	Публикация на русском языке
26-27 мая 1995 г. Пало Альто, шт. Калифорния. NSF Workshop on the Future of Database Systems Research «Lagunita meeting 2»	1. Phil Bernstein 2. Ron Brachman 3. Mike Carey 4. Rick Cattell 5. Hector Garcia-Molina 6. Laura Haas 7. Dave Maier 8. Jeff Naughton 9. Michael Schwartz 10. Pat Selinger 11. Avi Silberschatz 12. Mike Stonebraker 13. Jeff Ullman 14. Patrick Valduriez 15. Moshe Vardi 16. Jennifer Widom 17. Gio Wiederhold 18. Marianne Winslett 19. Maria Zemankova	1. Филипп Бернштейн 2. Рон Брахман 3. Майкл Кэри 4. Рик Каттел 5. Гектор Гарсиа-Молина 6. Лаура Хаас 7. Дейв Майер 8. Джефф Нотон 9. Майкл Шварц 10. Пат Селинджер 11. Эви Зильбершцац 12. Майк Стоунбрейкер 13. Джефф Ульман 14. Патрик Вальдурец 15. Мойше Варди 16. Дженифер Вайдом 17. Джо Видерхолд 18. Марианна Винслетт 19. Мария Земанкова	Avi Silberschatz, Mike Stonebraker, Jeff Ullman. Database Research: Achievements and Opportunities into the 21st Century. ACM SIGMOD Record, 25(1):52-63, 1996	Эви Зильбершцац, Майк Стоунбрейкер, Джефф Ульман. Базы данных: достижения и перспективы на пороге 21-го столетия. Новая редакция: Сергей Кузнецов, 2009 г., http://citforum.ru/database/classics/nfs_report/

Официально отчет о встрече опубликован в 1996 г. [4]. Неофициальная публикация доступна здесь: <http://ilpubs.stanford.edu:8090/81/1/1995-15.pdf>, дата обращения 26.03.2017. Имеется перевод отчета на русский язык. Вот что мне кажется наиболее интересным в этом отчете.

4.1 Рассредоточение информации

В 1996 г. уже всем было понятно, что человечество вступило в новую эру Всемирной Паутины. Участники встречи отмечали, что WWW – это распределенная среда, состоящая из автономных систем, узлы которой все чаще формируются как реляционные базы данных. Наличие этой среды заставляет переосмыслить многие концепции существующей технологии распределенных баз данных. Имелась в виду гипотетическая система, поддерживающая обработку запросов к Сети в целом и автоматически

определяющая, к каким Web-сайтам следует обратиться для выполнения запроса. Отмечались следующие направления требуемых исследований.

Учет и расчёты. Доступ к информации, хранимых в Web-сайтах, может быть платным. Гипотетическая система при обработке запроса пользователя должна учитывать, во сколько это может обойтись, и должна выбирать (может быть, в ущерб качеству ответа на запрос) тот набор Web-ресурсов, который окажет пользователю по карману. При этом, учитывая, что получение ответа на запрос не должно обходиться пользователю слишком дорого, сама эта служба учета и расчетов должна быть предельно экономичной.

Насколько мне известно, подобная проблема в масштабах Web не решена. Возможно (хотя и маловероятно), когда-нибудь за ее решение возьмется какая-нибудь крупная Internet-компания, например, Google. Однако имеются решения сходных проблем в контексте сервис-ориентированных архитектур (Data as a Service, [38]).

Безопасность и конфиденциальность. Как отмечалось в отчете, требуется разработка исключительно гибких систем аутентификации и авторизации, поддерживающих доступ на основе разнообразных «ролей», исполняемых пользователями (например, один и тот же индивид может выступать в роли лечащего врача некоторого пациента, в роли «врача вообще» или в роли частного лица). Тематика исследований и разработок в области безопасности и конфиденциальности в Internet чрезвычайно широка. Важность поддержки конфиденциальности данных многократно возросла в связи с появлением социальных сетей. Однако мне не приходилось слышать о защите конфиденциальности данных в масштабе Internet на основе ролей.

Репликация и согласование данных. Вот еще одна цитата из отчета: *из соображений эффективности данные часто реплицируются на нескольких узлах. Когда все эти узлы связаны сетью, можно поддерживать идентичность копий. Однако в ситуациях, когда связь нарушается, в копиях могут появиться различия. После восстановления связи должен включаться механизм согласования, который должен согласовать все копии и сформировать одну новую копию, отражающую все сделанные изменения. ... В новой информационной среде подобные ситуации становятся уже не исключением, а нормой.* Особенности распределенных систем категории NoSQL показывают, что достаточно эффективные механизмы согласования копий так и не появились. Следуя теореме CAP Эрика Брюера [39], разработчики этих систем обеспечивают высокий уровень доступности в ущерб обеспечения идентичности реплик.

Не полностью структурированные и неструктурированные данные. Участники встречи отдавали себе отчет, что у большинства данных в Web имеется в лучшем случае нечеткая, динамически изменяемая схема, а в худшем случае данные вообще не структурированы. Поэтому сообществу баз данных очень важно расширить механизмы индексации и другие средства поддержки поиска для хорошо структурированных данных и адаптировать их к неструктурированному миру Web. Естественно, такая работа была нужна, но

выполнило (и продолжает выполнять) ее не сообщество баз данных. Поиск информации в Web поддерживают универсальные и специализированные поисковые системы, основываясь на результатах активно развивающихся областей анализа текстов и семантического поиска. Из средств управления базами данных в лучшем случае используются СУБД категории NoSQL (в частности, системы «ключ-значение»).

4.2 Новые применения баз данных

Как говорится в отчете, *образовались новые важные области применения баз данных, и каждая из них представляет принципиально новую среду, к которой необходимо адаптировать технологии СУБД: интеллектуальный анализ данных (data mining), хранилища данных (data warehousing), репозитории данных (data repository).*

Интеллектуальный анализ данных. Отмечалась потребность в исследованиях по следующим направлениям: *методы оптимизации сложных запросов, включающих агрегацию и группирование; поддержка «многомерных» запросов, относящихся к данным, организованным в виде «куба», в ячейках которого находятся интересные данные; языки запросов очень высокого уровня, а также интерфейсы для поддержки пользователей, не являющихся экспертами, которым нужны ответы на нерегламентированные запросы.*

Здесь, прежде всего, нужно отметить имевшуюся на встрече путаницу между понятиями data mining и OLAP (On-Line Analytical Processing). Агрегация и группирование – это действия, требуемые для формирования многомерного куба, а анализ данных на основе их представления в виде куба – это OLAP. Так что в этом пункте речь идет об OLAP, а не о data mining.

Практически в то же время, когда проходила вторая встреча в Пало Альто, Джим Грей с коллегами придумали метод и соответствующие расширения языка SQL, позволяющие эффективно строить многомерный куб на основе табличного представления данных [40]. Эта работа оказала решающая воздействие на развитие технологии ROLAP (Relational OLAP), и ее принято считать одним из высших достижений Джима Грея. Эта же работа позволила создать новые методы оптимизации запросов с группировкой и агрегацией.

Годом позже в недрах компании Microsoft появился язык запросов в многомерном кубе MDX (MultiDimensional eXpressions) [41], основным разработчиком которого являлся Моша Пасуманский. Лично мне этот язык кажется еще более «ужасным», чем SQL, но распространено мнение, что он близок аналитикам. По крайней мере, являясь проприетарным языком, MDX реализован во всех системах, поддерживающих OLAP.

Близкие аналитикам интерфейсы, позволяющие на полунтуитивном уровне исследовать многомерные кубы, поддерживаются во всех OLAP-системах. Однако какой-либо стандартизации в этом направлении не видно.

Хранилища данных. Участники встречи считали, что для эффективного построения хранилищ данных требуется создать дополнительные

инструменты: *средства для создания насосов данных (data pump), т.е. модулей, функционирующих над средой источников данных и поставляющих в хранилище те изменения, которые существенны с точки зрения хранилища; методы «чистки данных» (data scrubbing), которые обеспечивают согласование данных, удаление элементов, соответствующих разным представлениям одного и того же а также удаление неправдоподобных значений; средства для создания и поддержки метасловаря, информирующего пользователей о способах получения данных.*

Поскольку физически интегрирующие данные из разных источников и сохраняющие «исторические» данные хранилища данных были и остаются незаменимым поставщиком данных для OLAP-аналитиков, их качественному построению в последние 20 лет уделяли серьезное внимание как вендоры SQL-ориентированных СУБД, так и независимые производители. В целом все средства, обеспечивающие извлечение данных из внешних источников, их преобразование к схеме хранилища данных и очистку, загрузку в хранилище данных, принято называть средствами ETL (Extract, Transform, Load). Относительно современное представление об ETL описано в Части IV [42] («научные» публикации на эту тему отсутствуют).

Конечно, в число средств ETL входит средство очистки данных. Однако, насколько я понимаю, очистка в основном предполагает приведение данных к единому формату и удаление избыточных данных. Менее формальная очистка требует выполнения семантического анализа загружаемых данных и оказывается слишком трудозатратной (речь идет об очень больших объемах данных). Про «насосы данных» в том смысле, в котором они упоминались в отчете, слышать не приходилось.

Отслеживание происхождения данных (data lineage) и сейчас считается важным средством. Исследования продолжаются, но практических результатов не видно.

Репозитории данных. Вот, что имелось в виду в отчете под термином *репозиторий*. *Приложения, относящиеся к категории репозитариев, характеризуются тем, что они предназначены для хранения и управления как данными, так и метаданными, т.е. информацией о структуре данных. Примеры репозитариев – базы данных для поддержки компьютерного проектирования, включая CASE (системы проектирования программного обеспечения), а также системы управления документами. Отличительная черта этих систем – частые изменения метаданных, характерные для любой среды проектирования.*

В 1990-е гг. технология репозитариев считалась перспективной (см., например, [43]). Однако впоследствии, как мне кажется, сообществу баз данных не удалось договориться с сообществом автоматизации проектирования, для которого, собственно, технология и предназначалась. Сегодня о репозиториях в этом смысле и не слышно.

5. Встреча в Кембридже: 1996

Четвертая встреча состоялась через год после третьей в Массачусетском технологическом институте 14-15 июня 1996 г. в рамках организованного ACM симпозиума Strategic Directions in Computing Research. Детали содержатся в приводимом фрагменте табл. 1.

Время и место встречи	Список участников на английском языке	Список участников на русском языке	Официальная публикация	Публикация на русском языке
14-15 июня 1996 г. MIT, Кембридж, шт. Массачусетс ACM Workshop on Strategic Directions in Computing Research «Cambridge meeting»	1. Jose Blakeley 2. Peter Buneman 3. Umesh Dayal 4. Tomasz Imielinski 5. Sushil Jajodia 6. Hank Korth 7. Guy Lohman, 8. Dave Lomet 9. Dave Maier 10. Frank Manola 11. Tamer Ozsu 12. Raghu Ramakrishnan 13. Krithi Ramamritham 14. Hans Schek 15. Avi Silberschatz 16. Rick Snodgrass 17. Jeff Ullman 18. Jennifer Widom, 19. Stan Zdonik	1. Хосе.Блейкли 2. Питер.Бьюнман 3. Умеш.Дайал 4. Томаш.Имилинский 5. Сушил.Джаджодиа 6. Хэнк.Корт 7. Гай.Лохман 8. Дейв.Ломе 9. Дейв.Майер 10. Френк.Манола 11. Тамер.Оззу 12. Раджу.Рамакришнан 13. Крийти.Рамамритан 14. Ганс.Шек 15. Эви.Зильбершца 16. Рик.Снодграсс 17. Джефф.Ульман 18. Дженифер.Вайдом 19. Стенли.Здоник	Avi Silberschatz, Stan Zdonik, et al. Strategic Directions in Database Systems: Breaking Out of the Box. ACM Computing Surveys, 28(4):764-778, 1996	А.Зильбершца, С.Здоник и др. Стратегические направления в системах баз данных Перевод: М.Р. Когаловский Новая редакция: Сергей Кузнецов, 2009 г., http://citforum.ru/database/classics/n_sf_report2/

Официально отчет о встрече опубликован в 1996 г. [5], ненамного позже отчета о предыдущей встрече. Предварительной неформальной публикации отчета не было, но в свободном доступе имеется текст официальной статьи: <https://www2.cs.arizona.edu/~rts/pubs/CompSurvDec96.pdf>, дата обращения 27.03.2017. Статья переведена на русский язык. Вот наиболее интересные прогнозы.

5.1 Расширяемость и компонентизация

Среди стратегически важных исследовательских проблем в отчете упоминаются следующие. *Нужно создать системы, которые дают возможность разработчику легко вводить новые типы данных, разработанные вне данной СУБД, которыми можно манипулировать внутри базы данных наравне с ее собственными полноправными типами. Нужно найти способы сделать архитектуру СУБД открытой таким образом, чтобы могли подключаться новые функциональные компоненты, и чтобы функциональные возможности системы базы данных могли конфигурироваться более гибкими способами в соответствии с потребностями приложений.*

Напомним, что встреча состоялась в середине 1996 г. В начале года компания Informix приобрела компанию Illustra вместе с ее технологий построения объектно-реляционных СУБД [30]. Насколько я понимаю, ко времени встречи

в Кембридже в Informix кипела работа по интеграции тогдашнего основного продукта компании Informix Dynamic Server и СУБД Illustra, чтобы к концу года можно было построить обещанный руководством компании Informix Universal Server. Стандарта SQL, включающего объектно-реляционные возможности, еще не было, и разработчики опирались на еще очень недоработанные драфты SQL3 [25]. В том же 1996 г. компания Sun Microsystems реализовала язык и виртуальную машину Java [44], компонентную модель разработки программного обеспечения Java Beans [45] и интерфейс Java с SQL-ориентированными базами данных JDBC [46].

Думаю, что в этом контексте нужно рассматривать первую часть прогноза. Конечно, нет и не было возможности экспортировать в базу данных разработанные вне ее пользовательские типы данных, чтобы можно было использовать их наравне с типами данных, определенными средствами SQL (например, определять столбцы таблиц). Но серверные программы (храняемые процедуры, функции и методы новых типов данных) можно программировать на разных языках, используя разные библиотеки, в том числе, и технологию Java Beans.

Что касается компонентной организации самих СУБД, то эта идея привлекала многих исследователей и до, и после встречи в Кембридже. На мой вкус, наиболее убедительной публикацией на эту тему является [47]. Идея внешне выглядит очень понятной и привлекательной. Универсальные СУБД рассчитаны на общий случай использования. В любой конкретной среде и при любой конкретной рабочей нагрузке многие составляющие СУБД просто не работают, а их наличие только замедляет обработку пользовательских запросов. Было бы логически правильно учитывать (желательно автоматически) особенности текущей среды использования и конфигурировать СУБД таким образом, чтобы в ней не оставались ненужные в данное время компоненты.

Наверное, технически можно построить такую компонентную, автоматически конфигурируемую СУБД. Но сложность состоит в том, что чем более структурно строится любая сложная система (ОС, СУБД и т.д.), тем больше внутри нее требуется взаимодействия компонентов, и эти взаимодействия сами снижают производительность системы. Многие программные системы изначально проектировались высоко структурным образом как набор относительно автономных взаимодействующих компонентов, а потом в целях повышения эффективности приводились к практически монолитной архитектуре.

Интересно, что примерно те же соображения о недостатках универсальных СУБД привели в начале XXI века к идее о потребности перехода к специализированным СУБД [48].

5.2 Оптимизация запросов

По этому поводу в отчете говорится, что *могут измениться критерии оптимизации. Например, пользователи могут предпочесть получать ответы на свои запросы с меньшей скоростью, точностью и полнотой, если это будет стоить заметно дешевле.*

Оптимизация запросов – эта одна из наиболее сложных функций СУБД (и сейчас, и 20 лет тому назад). Здесь не место вдаваться в технические подробности (их очень много, и они нетривиальны). На мой взгляд, сегодня общее представление о сути оптимизации запросов лучше получать не из академических публикаций, а из документации вендоров (например, [49]).

Грубо говоря, работа оптимизатора запросов состоит из трех фаз: преобразование запроса, генерация планов выполнения запроса, оценка планов и выбор наиболее дешевого плана. На вход оптимизатора поступает внутреннее представление запроса, полученное синтаксическим анализатором. На первой фазе это представление преобразуется в семантически эквивалентное (непроцедурное) представление, обладающее предположительно улучшенным качеством (например, запросы с вложенными подзапросами приводятся к запросам с соединениями). На второй фазе, руководствуясь набором заложенных в СУБД стратегий и методов выполнения отдельных операций низкого уровня, оптимизатор строит набор процедурных планов выполнения запроса. Для сложных запросов этот набор потенциально может быть очень большим, и задачей оптимизатора является сохранение его осмысленного размера, используя эвристики, отбрасывающие предположительно худшие планы. Наконец, на третьей фазе производится оценка каждого оставленного в наборе плана, и выбирается план с наименьшей стоимостью.

В работе оптимизатора используются два фундаментальных предположения: запрос нужно выполнить полностью и точно; база данных сохраняется в традиционной дисковой памяти на устройствах с подвижными головками. Отказ от этих предположений может потребовать внесение принципиальных изменений на каждой фазе работы оптимизатора. Поэтому изменять критерии оптимизации запросов – дело накладное и сложное. Производители СУБД идут на это с большой неохотой. Прогноз можно считать несбывшимся.

5.3 Интеллектуальный анализ данных внутри СУБД

Участники встречи считали, что *классификация и кластеризация могут рассматриваться как случайные запросы, для которых необходимы новые семейства языков запросов. К числу исследовательских задач в этой области относится разработка адекватного набора простых примитивов запросов и нового поколения методов оптимизации запросов.*

В 1990-е гг. тема нагрузки SQL-ориентированных СУБД функциями интеллектуального анализа данных была сравнительно популярна в исследовательском сообществе баз данных [50]. Тогда это мотивировалось тем, что технология баз данных уже переросла традиционные потребности бизнес-

приложений, она способна поддерживать новые классы приложений, в том числе и аналитических. Один из вопросов, который занимал тогдашних исследований, состоял в том, не повлияет ли выполнение интеллектуального анализа данных внутри СУБД на эффективность систем при обработке запросов?

Сегодня ведущие поставщики SQL-ориентированных СУБД (например, компании Oracle и Microsoft) поставляют готовые решения data mining внутри баз данных, а также предоставляют средства разработки дополнительных серверных аналитических механизмов. Сегодняшняя мотивировка серверного интеллектуального анализа данных – это потребность в переносе обработки данных как можно ближе к месту их хранения. Серверная аналитика позволяет значительно сократить объем данных, передаваемых от серверов баз данных на рабочие станции, снизить уровень требований к аппаратному оснащению рабочих станций.

Вместе с тем, современные средства и механизмы интеллектуального анализа данных все более ориентируются на обработку неструктурированных данных (текст, изображения, видео-, аудиоданные и т.д.). Преимущества СУБД же в основном связаны с управлением структурированными данными. Трудно сказать, насколько востребованы сегодня средства интеллектуального анализа данных, встроенные в SQL-ориентированные СУБД. К сожалению, часто наблюдается обратная картина – в приложениях data mining вообще не используются СУБД.

А прогноз о развитии языков запросов и их оптимизаторов для поддержки интеллектуального анализа данных в целом не оправдался.

6. Асиломарская встреча: 1998

Пятая встреча состоялась 19-21 августа 1998 г. в парке Асиломар неподалеку от г. Монтерей, шт. Калифорния. Во фрагменте табл. 1 описаны детали встречи.

Время и место встречи	Список участников на английском языке	Список участников на русском языке	Официальная публикация	Публикация на русском языке
19-21 августа 1998 г. Асиломар, г. Пасифик Гров, шт. Калифорния «Asilomar meetong»	1. Phil Bernstein 2. Michael Brodie 3. Stefano Ceri 4. David DeWitt 5. Mike Franklin 6. Hector Garcia-Molina 7. Jim Gray 8. Jerry Held 9. Joe Hellerstein 10. H. V. Jagadish 11. Michael Lesk 12. Dave Maier 13. Jeff Naughton 14. Hamid Pirahesh 15. Mike Stonebraker 16. Jeff Ullman	1. Филипп Бернштейн 2. Майкл Броуди 3. Стефано Чери 4. Дэвид Девитт 5. Майк Франклин 6. Гектор Гарсиа-Молина 7. Джим Грей 8. Джерри Хелд 9. Джо Хеллерштейн 10. Х.В. Ягадиш 11. Майкл Леск 12. Дейв Майер 13. Джефф Нотон 14. Хамид Пирамеш 15. Майк Стонбрейкер 16. Джефф Ульман	Phil Bernstein, Michael Brodie et al. The Asilomar Report on Database Research. ACM SIGMOD Record, 27(4):74-80, 1998	Филипп Бернштейн, Майкл Броуди и др. Асиломарский отчет об исследованиях в области баз данных. Перевод: Сергей Кузнецов, 1999, http://citforum.ru/database/digest/asil_01.shtml

Официальный отчет о встрече был опубликован в том же 1998 г. [6]. Предварительной неофициальной публикацией можно считать <https://arxiv.org/html/cs/9811013>, дата обращения 30.03.2017. Среди прогнозов Асиломарского отчета наиболее интересны следующие.

6.1 Системы управления базами данных в стиле «plug and play»

Участники Асиломарской встречи отмечали, что *в связи с ростом относительной стоимости человеческого фактора в компьютерных системах требуется, чтобы будущие компьютерные системы стали полностью автоматическими: автоинсталлируемыми, автоуправляемыми, авторемонтируемыми и автопрограммируемыми. В том числе требуется обеспечение самонастраиваемости систем баз данных, т.е. удаление массы параметров настройки производительности, которые должны определять пользователи в текущих продуктах. В частности, должны появиться методы автоматического выбора индексов.*

Важность проблемы самонастраиваемости систем баз данных в XXI веке осознается всеми крупными вендорами SQL-ориентированных СУБД. По всей видимости, первой компанией, в которой начались масштабные исследования в этом направлении, была Microsoft (проект AutoAdmin под руководством Сураджиты Чаудхари начался в 1996 г [51, 52]).

В начале этого проекта было разработано средство Index Tuning Wizard, которое анализировало текущую рабочую нагрузку СУБД (операции выборки данных, вставки, модификации и удаления строк таблиц) и устанавливала, при каком наборе индексов на соответствующих таблицах соответствующий набор операций обрабатывался бы СУБД в целом наиболее эффективно. При анализе рабочей нагрузки уже тогда использовались методы data mining. В дальнейшем в этом средстве стали учитываться не только индексы, но и материализованные представления и другие объекты базы данных, наличие или отсутствие которых влияет на эффективность обработки операций.

Имеющиеся сегодня в передовых СУБД средства самонастройки в основном направлены на то, чтобы помочь администраторам баз данных настроить физическую схему базы данных, дабы она наилучшим образом соответствовала особенностям текущей среды использования. Кроме того, в системы широко внедряются средства мониторинга, результаты которых позволяют администраторам распознать как собственные недочеты при настройке базы данных, так и слабые места СУБД (например, оптимизатора запросов). Кроме работ, ведущихся в Microsoft, в этой области заметны усилия исследователей и разработчиков компании Oracle [53].

Конечно, имеющиеся и ожидаемые средства этого рода помогают администраторам баз данных и настройщикам приложений. Однако совсем не факт, что их наличие позволяет существенно сократить число «ручек

управления», предоставляемых администраторам со стороны СУБД. Автоматическую настройку баз данных без участия человека-эксперта по-прежнему не поддерживает ни одна система, а от администраторов требуется еще более высокая квалификация. Так что прогноз нельзя считать сбывшимся, хотя он остается путеводной звездой для производственных исследователей.

6.2 Переосмысление традиционной архитектуры систем баз данных

В отчете утверждалась *потребность в переосмыслении традиционной архитектуры систем баз данных в свете появления среды, которая будет доступна в 2010-ом году.*

Эти утверждения созвучны названию статьи [34]: «Конец архитектурной эпохи, или Наступило время полностью переписывать системы управления данными». Да и вышла эта статья в 2007 г., аккурат к концу первого десятилетия XXI века. Однако при всем уважении к авторам [34] нельзя сказать, что они предлагают полностью переосмысленные архитектуры СУБД. Они переосмысливают их только в связи с переходом от архитектур универсальных СУБД к архитектурам специализированных систем. Компоненты архитектур остаются известными и традиционными.

Да в общем-то и оснований для полного пересмотра архитектуры в 2000-е гг. не было. Имелись в основном количественные, а не качественные изменения (существенно увеличенные размеры основной памяти, кластеры с очень большим числом узлов и т.д.). Неизменной оставалась опора SQL-ориентированных СУБД на жесткие диски с подвижными головками. (Мне близка следующая цитата из Асиломарского отчета: «в будущем могут оказаться неуместными архитектуры, “скрывающие наличие подвижных магнитных головок”, такие как RAID 5» – с позиций оптимизации запросов мне всегда были подозрительны устройства внешней памяти, скрывающие свою организацию.)

Зато, по моему мнению, это время наступило сейчас (и, видимо, продлится в следующем десятилетии, потому что новые архитектуры быстро не рождаются). Вот что может действительно повлиять на архитектуру будущих СУБД (новые аппаратные возможности перечисляются в порядке возрастания их влияния на архитектуру СУБД).

Флэш-память и твердотельные диски. Твердотельная внешняя память (Solid-State Disk, SSD) обладает принципиально иными характеристиками, чем традиционная дисковая память на дисковых устройствах с подвижными головками (Hard Disk, HD). Основным отличием является то, что у устройств SSD отсутствуют механически перемещаемые магнитные головки, и они действительно являются устройствами прямого доступа – время доступа к любому блоку данных в SSD одно и то же. И даже без учета времени перемещения головок у HD SSD показывают время обмена, в несколько раз меньшее, чем у HD.

Раньше SSD проигрывали HD по максимально доступному объему, стоимости в расчете на гигабайт данных и износоустойчивости. В настоящее время уже доступны SSD емкостью в несколько терабайт. Показатели стоимости и износоустойчивости пока уступают соответствующим показателям HD, но со временем и они будут улучшены.

Естественно, возникает желание перевести имеющиеся СУБД на платформу SSD. Но просто это не делается. Особенности SSD заставляют пересмотреть алгоритмы, используемые во многих компонентах SQL-ориентированных СУБД, а может быть и архитектуру СУБД в целом. Например, при переходе от SSD к HD перестают эффективно работать методы кэширования блоков базы данных в основной памяти [54], требуется их принципиальная переделка.

Безусловно, переход к SSD требует значительной переделки оптимизатора запросов, начиная, по-видимому, с эвристик «отсеивания» заведомо негодных планов запросов. Более того, в наборе планов для данного запроса могут появиться планы, которые вообще не генерировались при использовании HD. Полностью меняется компонент оценки стоимости плана, поскольку теперь обмены с внешней памятью будут стоить гораздо дешевле, и эта стоимость не будет зависеть от расположения блока данных во внешней памяти.

У меня имеется ощущение, что основные вендоры SQL-ориентированных СУБД опасаются этих изменений, затрагивающих самые внутренние части СУБД, и поэтому используют SSD, в основном пренебрегая их отличиями от HD.

Энергонезависимая основная память. Этот вид памяти, называемой non-volatile memory (долговременной, или энергонезависимой памятью), а также storage class memory (память класса хранения данных), наконец-то становится доступным. Еще 30 лет тому назад использование энергонезависимой памяти предполагалось в проекте системы хранения Postgres [55]. Стоунбрейкер хотел использовать такую память (в небольшом объеме) для хранения части кэша блоков базы данных и журнала. Конечно, тогда энергонезависимую память взять было негде, как, собственно, и в следующие десятилетия вплоть до нашего времени. Сейчас память класса хранения данных стала реальностью. Имеется несколько вариантов физического исполнения такой памяти, но все они обеспечивают прямую адресацию и сохранность данных после отключения электропитания.

Конечно, очень заманчиво использовать энергонезависимую память в СУБД. Сегодня достаточно популярны СУБД, хранящие данные в традиционной основной памяти (in-memory DBMS). Но эти системы не могут обойтись без дисковой памяти, потому что для всех транзакций, изменяющих данные, должна обеспечиваться гарантированная сохранность результатов (durability), т.е. эти результаты так или иначе должны быть отображены во внешнюю память. По этой причине системы in-memory обычно обеспечивают существенно большую производительность, чем традиционные СУБД, для только читающих транзакций. (Исключением является массивно-параллельная

архитектура H-Store/VoltDB [34, 36], в которой durability транзакций поддерживается за счет репликации данных.)

Если вся система хранения данных в СУБД поддерживается в энергонезависимой памяти, то внешняя память перестает быть необходимой. Но для построения таких СУБД нужны кардинально другие подходы к представлению данных, организации индексов, управлению транзакциями, журнализации, оптимизации запросов и т.д. Что-то можно позаимствовать из арсенала СУБД, хранящих данные в традиционной памяти, но очень многое, включая общую архитектуру системы нужно изобретать заново. В последние годы соответствующие исследования начались [56-57], но как показывают публикации, они находятся на начальной стадии, и предстоит еще много работы, пока удастся получить OLTP-ориентированную СУБД, работающую со скоростью основной памяти.

Массивно-мультипоточные архитектуры компьютеров. Сегодняшние компьютеры рассчитаны на то, что выполняемые в них программы соблюдают принцип локальности. Это означает, что в любой момент выполнения любого процесса ему достаточно обеспечить некоторый ограниченный набор команд и данных (рабочий набор), причем если процесс обладает рабочим набором WS в момент времени T , то с большой вероятностью этот рабочий набор сохранится и в момент времени $T+t$ для некоторого значения t . Принцип локальности позволяет поддерживать в процессорах аппаратный кэш, и именно наличие этого кэша дает возможность сгладить разрыв в скорости между процессором и основной памятью.

Однако существуют классы приложений, в которых программам не свойственен принцип локальности (приложения биоинформатики, анализа социальных сетей и т.д.). При работе таких приложений наличие кэша никак не способствует эффективности, и они работают, по сути, со скоростью основной памяти. Идея аппаратной архитектуры для поддержки таких приложений появилась в конце прошлого десятилетия в компании Cray Inc. [58].

Суть идеи состоит в том, что нужно сделать массивно-многопоточный процессор, поддерживающий на аппаратном уровне десятки тысяч потоков с общим доступом к основной памяти. Кэш в процессоре не поддерживается, но для каждого потока обеспечивается собственный набор регистров. Если в данный момент времени выполняется поток $Tr1$, и в нем потребовался обмен с основной памятью, то он аппаратно блокируется, и запускается один из потоков Tri , у которого все требуемые данные находятся в регистрах. При достаточно большом числе потоков готовый к выполнению поток всегда найдется, и многопоточная программа будет в целом выполняться со скоростью процессора.

Для достижения таких результатов нужно научиться распараллеливать программу на очень большое число небольших (несколько десятков команд) потоков. Это очень сложная задача, отсутствие решение которой тормозит развитие массивно-мультипоточных архитектур. Но, по моему мнению, потенциальная возможность построения такой архитектуры должна

стимулировать исследования в сообществе баз данных. Предположим, что массивно-мультиточный компьютер оснащен энергонезависимой основной памятью, и эта память используется для хранения баз данных. СУБД работает на некотором внешнем компьютере с традиционной архитектурой и при компиляции операторов SQL распараллеливает их на большое число потоков. Это гораздо более реально, чем для произвольной программы, потому что SQL – декларативный язык.

Конечно, это очень непростая задача, но если ее решить, мы сможем получить системы баз данных, обрабатывающие запросы со скоростью кэша.

7. Лоуэллская встреча: 2003

Следующая, шестая встреча исследователей в области баз данных состоялась 4–6 мая 2003 г. в Лоуэлле, шт. Массачусетс. Фрагмент табл. 1 описывает детали встречи.

Время и место встречи	Список участников на английском языке	Список участников на русском языке	Официальная публикация	Публикация на русском языке
4–6 мая 2003 г. Лоуэлл, шт. Массачусетс «Lowell meeting»	1. Serge Abiteboul 2. Rakesh Agrawal 3. Phil Bernstein 4. Mike Carey 5. Stefano Ceri 6. Bruce Croft 7. David DeWitt 8. Mike Franklin 9. Hector Garcia Molina 10. Dieter Gawlick 11. Jim Gray 12. Laura Haas 13. Alon Halevy 14. Joe Hellerstein 15. Yannis Ioannidis 16. Martin Kersten 17. Michael Pazzani 18. Mike Lesk 19. David Maier 20. Jeff Naughton 21. Hans Schek 22. Timos Sellis 23. Avi Silberschatz 24. Mike Stonebraker 25. Rick Snodgrass 26. Jeff Ullman 27. Gerhard Weikum 28. Jennifer Widom 29. Stan Zdonik	1. Серж Абитебуль 2. Ракеш Агравал 3. Филипп Бернштейн 4. Майк Кэри 5. Стефано Чери 6. Брюс Крофт 7. Дэвид Девиитт 8. Майк Франклин 9. Гектор Гарсиа-Молина 10. Дитер Гавлик 11. Джим Грей 12. Лаура Хаас 13. Элон Хэлеви 14. Джо Хеллерштейн 15. Янис Ионнидис 16. Мартин Керстен 17. Майкл Паззани 18. Майк Леск 19. Дэвид Мейер 20. Джефф Нотон 21. Ганс Шек 22. Тимос Селлис 23. Эви Зильбершац 24. Майкл Стоунбрейкер 25. Рик Снодграсс 26. Джефф Ульман 27. Герхард Вейкум 28. Дженифер Вайдом 29. Стен Здоник	Serge Abiteboul, Rakesh Agrawal et al. The Lowell Database Research Self-Assessment. Communications of the ACM, 48(5):111-118, 2005	Сергей Кузнецов. Крупные проблемы и текущие задачи исследований в области баз данных, 2005, http://citforum.ru/database/articles/problems/

Официальная публикация Лоуэллского отчета [7] появилась в 2005 г., предварительная неофициальная публикация доступна на <http://jimgray.azurewebsites.net/lowell/lowelldatabase-research-self-assessment.pdf>, дата обращения 1 апреля 2017 г. В [8] основные положения отчета пересказываются и комментируются на русском языке. Вот наиболее интересные прогнозы.

7.1 Интеграция текста, данных, кода и потоков

В отчете утверждается, что *пора прекратить встраивать новые конструкции в старую реляционную архитектуру. Нужно переосмыслить базовую архитектуру СУБД с целью поддержки структурированных данных; текстовых, пространственных, темпоральных и мультимедийных данных; процедурных данных, т.е. типов данных и инкапсулирующих их методов; триггеров; потоков и очередей данных как равноправных компонентов первого сорта внутри архитектуры СУБД (как на уровне интерфейсов, так и на уровне реализации). Для исследовательского сообщества требуется выработка новой системы понятий.*

На мой взгляд, никакой «новой системы понятий» не появилось. Да и попыток переосмысления базовой архитектуры универсальных СУБД за прошедшие годы видно не было. С одной стороны, основные вендоры коммерческих СУБД продолжали наращивать функциональные возможности своих продуктов, стараясь как можно меньше изменять ядро систем (т.е., по сути, делая новые компоненты объектами «второго сорта»). С другой стороны, с 2005 г. на мир баз данных воздействует слоган Майкла Стоунбрейкера «Один размер непригоден для всех» [48], и с разным успехом было выполнено несколько проектов специализированных СУБД, в которых на первое место выходили именно те понятия, на поддержку которых ориентировалась система.

В качестве поучительного примера попытки расширить число объектов «первого сорта» в универсальных СУБД стоит вспомнить сравнительно недавнюю историю поддержки XML в базах данных. Спецификация XML (eXtensible Markup Language) появилась в начале 1998 г. Язык в основном предназначался для представления сообщений, передаваемых в Internet. Представлялось, что таких сообщений будет очень много, полезно уметь их сохранять и производить поиск в получаемых коллекциях XML-документов. Это было толчком к разработке специализированных XML-ориентированных СУБД с собственными системами хранения и поддержкой языка запросов XQuery. Первыми эту работу начали компания Software AG (Tamino [59]) и ИСП РАН (Sedna [60]).

Tamino и Sedna были уже сравнительно работоспособными системами, когда в игру вступили IBM и Oracle. В SQL-ориентированной СУБД XML с собственным языком запросов поддерживать не так уж просто, и вендорам, фактически, пришлось реализовать отдельную среду хранения XML, процессор и оптимизатор языка XQuery, а затем интегрировать все это новое хозяйство со

средой SQL, обеспечив возможность встраивания XQuery в SQL, SQL – в XQuery и т.д. Потребовалась большая и дорогостоящая работа, чтобы сделать XML и XQuery «полноправными жителями» SQL-ориентированной СУБД. По моим понятиям, обе компании выполнили эту работу прекрасно. Но ко времени достижения готовности к использованию таких интегрированных систем интерес к XML затух. Новым фаворитом стал язык JSON, а результаты проделанной работы (как в области создания специализированных XML-ориентированных систем, так и в направлении разработки интегрированных СУБД) оказались во много не востребованными.

Но на создание специализированных систем было затрачено меньше сил и средств. Так что стоит подумать о целесообразности новой системы понятий для универсальных СУБД.

7.2 Объединение информации

Участники встречи отмечали, что *в Internet парадигма ETL не приемлема. Теперь требуется производить интеграцию информации между несколькими предприятиями. В результате потребуется интеграция, возможно, миллионов информационных источников «на лету».*

В такой общей постановке проблема не решена (скорее всего, она и вовсе неразрешима). Реалистичный подход был предложен в 2005 г. в [61]. В этой статье утверждалось, что унифицированный подход к интеграции данных невозможен и не нужен. Должен обеспечиваться тот уровень интеграции, который требуется конкретному типу приложений. Например, для выполнения интеллектуального анализа данных достаточно физически собрать данные из разных источников без какого-либо их преобразования, а для классического OLAP требуется построение хранилища данных с полной поддержкой ETL. К сожалению, на практике красивая идея пространств данных пока востребована не была.

Если продолжать говорить про ETL и OLAP, то в последние годы сравнительно жизнеспособной является идея виртуальных хранилищ данных [62], в которых поддерживаются обширные метаданные и инкрементально обновляемые агрегаты, а доступ к фактам производится «на лету» в соответствующих внешних источниках. В таких «хранилищах данных» не поддерживаются исторические данные, и возможности OLAP ограничены. За это и другие объективные недостатки виртуальные хранилища данных достаточно резко ругает классик технологии хранилищ данных Билл Инмонн [63].

8. Клермонтская встреча: 2008

Седьмая встреча состоялась 29-30 мая 2008 г. в гостинице Клермонт Ресорт в Беркли, шт. Калифорния. Подробности приведены во фрагменте табл. 1.

Время и место встречи	Список участников на английском языке	Список участников на русском языке	Официальная публикация	Публикация на русском языке
29-30 мая 2008 г. Беркли, шт. Калифорния, Клермонт Ресорт «Claremont meeting»	1. Rakesh Agrawal 2. Anastasia Ailamaki 3. Philip A. Bernstein 4. Eric A. Brewer 5. Michael J. Carey 6. Surajit Chaudhuri 7. AnHai Doan 8. Daniela Florescu 9. Michael J. Franklin 10. Hector Garcia Molina 11. Johannes Gehrke 12. Le Gruenwald 13. Laura M. Haas 14. Alon Y. Halevy 15. Joseph Hellerstein 16. Yannis E. Ioannidis 17. Hank F. Korth 18. Donald Kossmann 19. Samuel Madden 20. Roger Magoulas 21. Beng Chin Ooi 22. Tim O'Reilly 23. Raghu Ramakrishnan 24. Sunita Sarawagi 25. Michael Stonebraker 26. Alexander S. Szalay 27. Gerhard Weikum	1. Ракеш Агравал 2. Анастасия Айламаки 3. Филипп Бернштейн 4. Эрик Брюер 5. Майкл Кэри 6. Сураджит Чаудхари 7. Анхай Доан 8. Даниела Флореску 9. Майкл Франклин 10. Гектор Гарсиа-Молина 11. Иоханнес Герке 12. Ле Грюнвальд 13. Лаура Хаас 14. Элон Хэлеви 15. Джозеф Хеллерштейн 16. Янис Ионнидис 17. Хэнк Корт 18. Дональд Коссмманн 19. Сэмюэль Мэдден 20. Роджер Магулас 21. Бен Чин Ой 22. Тим О'Рейли 23. Раджу Рамакришнан 24. Сунита Сарагави 25. Майкл Стоунбрейкер 26. Александр Шалай 27. Герхард Вейкум	Rakesh Agrawal, Anastasia Ailamaki, et al. The Claremont Report on Database Research. Comm. of the ACM, 52(6):56-65, 2009.	Ракеш Агравал, Анастасия Айламаки др. Клермонтский отчет об исследованиях в области баз данных. Пересказ и комментарии: Сергей Кузнецов, 2008, http://citforum.ru/database/articles/claremont_report/

Официальный отчет о встрече опубликован в [9] в 2009 г., предварительная версия доступна на специальном сайте <http://db.cs.berkeley.edu/claremont/>, дата обращения 2 апреля 2017 г. (на этом сайте также размещены слайды участников встречи – нечто вроде их представления друг другу, очень интересно!). На русском языке имеется мой пересказ отчета с комментариями. На встрече отмечался ряд факторов, влияющих на направления исследований в области баз данных.

8.1 Повышение ажиотажа вокруг Больших Данных

По мнению участников встречи, наличие проблемы Больших Данных *приводит к быстрому росту числа пользователей традиционных систем управления базами данных (СУБД), а также стимулирует разработку новых специализированных решений управления данными на основе упрощенных компонентов. Повсеместное использование больших данных приводит и к возрастанию числа разработчиков технологий управления данными, что, несомненно, вызовет коренную реорганизацию этой области.*

С этим мнением участников Клермонтской встречи нельзя не согласиться, хотя уже в 2008 г. оно выглядело не прогнозом, а скорее наблюдением за происходящим. К этому времени уже существовали специализированные массивно-параллельные аналитические СУБД Vertica [35], Greemplum и Aster Data [12], специализированная потоковая система StreamBase [48], был готов прототип специализированной массивно-параллельной транзакционной СУБД H-Store [34] и велась подготовка ее коммерциализации (VoltDB [36]) и т.д. Разработка новых специализированных решений ведется и в настоящее время (см. например, [64]).

Интересно, что почти все перечисленные компании были поглощены более крупными и заслуженными компаниями, в основном не специализирующимися на производстве продуктов управления базами данных: Vertica была поглощена в 2011 г. компанией Hewlett-Packard; Greemplum – в 2010 г. компанией EMC; Aster Data – в 2011 г. компанией Teradata; StreamBase – в 2013 г. компанией TIBCO. Некоторые из этих продуктов видны за пределами их новых владельцев, некоторые – нет, т.е. несмотря на использование передового подхода Майкла Стоунбрейкера мы имеем всего лишь ряд удачных стартапов, а реальная технология управления базами данных по-прежнему в руках традиционных гигантов.

Что касается второй части наблюдения авторов отчета, то уже в 2008 г. направление NoSQL развивалось небывалыми в истории баз данных темпами. Быстро возрастало число реализованных (почти всегда с открытыми исходными кодами) систем, в которых отрицались SQL и ACID-транзакции. В 2006 г. начался проект Hadoop для открытой реализации MapReduce. Сегодняшний список нереляционных, распределенных, горизонтально масштабируемых СУБД с открытыми текстами насчитывает более 230 систем [66], Hadoop широко используется, развивается и совершенствуется.

Непонятно, означает ли это коренную реорганизацию области баз данных? Трудно сказать. Сегодня сообщество NoSQL сторонится как исследовательской работы, так и традиционного исследовательского сообщества баз данных. Уже несколько лет проводятся специализированные конференции, посвященные исключительно проблематике NoSQL, но если посмотреть на сайты этих конференций, видно, что они совсем не такие, как VLDB, Data Engineering, SIGMOD и т.д. – конференции, на которых исследователи рассказывают о полученных результатах. Они больше похожи на конференции, которые вендоры программных систем проводят для своих пользователей.

Не знаю, может ли технология развиваться без исследований. Не зайдет так в тупик направление NoSQL?

8.2 Изменения в архитектуре компьютерных систем

В отчете отмечаются следующие архитектурные изменения. *На макроуровне фундаментальным изменения в архитектуре программного обеспечения сулит развитие «облачных» компьютерных служб. На микроуровне в компьютерных архитектурах закон Мура теперь трактуется в пользу не повышения*

тактовой частоты микропроцессоров, а увеличения числа процессорных ядер и потоков управления в одном кристалле. Основные изменения в технологии хранения данных относятся к иерархии памяти в связи с доступностью большего числа кэшей увеличенного объема на одном кристалле, все более дешевой основной памяти большого объема и флэш-памяти.

С утверждением насчет *макроуровня* в целом нельзя не согласиться. Модель «Программное обеспечение как услуга» (Software As A Service, SAAS) в большом числе случаев позволяет пользователям избавиться от потребности в установке, администрировании и поддержке программного обеспечения. Конечно, это относится и к облачным СУБД. Но с ними связана проблема, на которую, как мне кажется, в сообществе облачных вычислений (да и в сообществе баз данных) закрывают глаза.

Основой облачных вычислений является виртуализация. Пользователь запрашивает у поставщика облачных услуг услугу с требуемыми характеристиками (на основе Соглашение об уровне предоставления услуги – Service Level Agreement, SLA), и его не касается, каким образом, за счет каких физических ресурсов эта услуга будет оказываться. Виртуализация на всех уровнях, от уровня IAAS (инфраструктура как услуга) до уровня SAAS предполагает, в частности, что облачная СУБД работает на виртуальном сервере, оснащенный виртуальной системой хранения.

В то же время, оптимизатор запросов этой облачной СУБД полагает, что СУБД по-прежнему работает с реальными магнитными дисками с подвижными головками и выбирает планы запросов, выполнение которых потребует меньшего числа обменов с дисковыми устройствами. Результаты могут оказаться непредсказуемыми и очень неприятными для пользователей. Я вижу только один реальный путь к устранению этой проблемы – при запросе услуги облачной СУБД обеспечивать ей реальную, а не виртуальную аппаратуру с характеристиками, определяемыми пользователями. Возможно, дело к этому и идет, но подтверждающие это публикации отсутствуют.

Что касается *микроуровня*, то распараллеливание запросов в симметричных мультипроцессорных системах (системах с общей основной памятью), к которым относятся и современные многоядерные и частично многопоточные компьютеры, поддерживается в развитых SQL-ориентированных СУБД не первый десяток лет. Реальной проблемой является то, что СУБД не может обеспечить горизонтальную масштабируемость при возрастании числа ядер (или нитей) в процессоре из-за ограничений параллельного доступа к основной памяти.

Некоторую надежду на возможность решения этой проблемы дает публикация [65]. В ней предлагается эскиз архитектуры СУБД с хранением данных в основной памяти, вроде бы обеспечивающей горизонтальную масштабируемость. Общая идея состоит в том, что в системе с n ядрами выполняется один процесс СУБД с n потоками, жестко привязанными к ядрам. В каждой нити выполняется вся СУБД целиком, как в массивно-параллельных СУБД без общих ресурсов, а разделение данных выполняется за счет механизма

виртуальной памяти. Потоки взаимодействуют путем обмена сообщениями на основе наличия общей памяти. При потребности заново разделить базу данных перепись данных не требуется, изменяется лишь таблица страниц виртуальной памяти. Компиляция и оптимизация запросов производятся во внешнем компьютере.

Наконец, свои соображения относительно средств долговременного хранения данных я уже изложил в 6.2. К этому можно добавить лишь то, что доступность больших объемов основной памяти активизировала разработки СУБД класса in-memory.

9. Бекманская встреча, 2013

Последняя к настоящему времени встреча прошла 14-15 октября 2013 г. в Бекманском центре университета в г. Ирвин, шт. Калифорния. Подробности в приводимом фрагменте табл. 1.

Время и место встречи	Список участников на английском языке	Список участников на русском языке	Официальная публикация	Публикация на русском языке
14-15 октября 2013 г. Ирвин, шт. Калифорния, Бекманский центр университета в Ирвине «Beckman meeting»	1. Daniel Abadi 2. Rakesh Agrawal 3. Anastasia Ailamaki 4. Magdalena Balazinska 5. Philip A. Bernstein 6. Michael J. Carey 7. Surajit Chaudhuri 8. Jeffrey Dean 9. AnHai Doan 10. Michael J. Franklin 11. Johannes Gehrke 12. Laura M. Haas 13. Alon Y. Halevy 14. Joseph Hellerstein 15. Yannis E. Ioannidis 16. H.V. Jagadish 17. Donald Kossmann 18. Samuel Madden 19. Sharad Mehrotra 20. Tova Milo 21. Jeffrey F. Naughton 22. Raghuram Ramakrishnan 23. Volker Markl 24. Christopher Olston 25. Beng Chin Ooi 26. Christopher Re 27. Dan Suciu 28. Michael Stonebraker 29. Todd Walter 30. Jennifer Widom	1. Дэниел Абади 2. Ракеш Агравал 3. Анастасия Айламаки 4. Магдалена Балазинска 5. Филип А. Берштейн 6. Майкл Дж. Кэри 7. Сураджит Чаудхари 8. Джеффри Дин 9. Анхай Доан 10. Майкл Дж. Франклин 11. Йоханнес Герке 12. Лаура М. Хаас 13. Элон И. Хэлеви 14. Джозеф Хеллерштейн 15. Янис Е. Ионнидис 16. Х.В. Ягадиш 17. Дональд Коссмани 18. Самуэль Мэдден 19. Шарад Мехротра 20. Това Мило 21. Джеффри Нотон 22. Раджу Рамакришнан 23. Волкер Маркл 24. Кристофер Олстон 25. Бен Чин Ой 26. Кристофер Ре 27. Дан Сучиу 28. Майкл Стоунбрейкер 29. Тодд Валтер 30. Джерифер Вайдом	Daniel Abadi, Rakesh Agrawal et al. The Beckman Report on Database Research. SIGMOD Record, September, 43(3):61-70, 2014	Дэниел Абади, Ракеш Агравал и др. Бекманский отчет об исследованиях в области баз данных. Перевод: Сергей Кузнецов, 2017

Официальная публикация появилась в 2014 г., неофициальная доступна на сайте встречи <https://beckman.cs.wisc.edu/>, дата обращения 2 апреля 2017 г. На этом сайте имеются слайды с представлениями участников и их групповой фотографией (рис. 1). Отчет переведен на русский язык.



Рис. 1. Групповая фотография участников Бекманской встречи

Pic. 1. The group photo of the Beckman meeting participants

(from <https://beckman.cs.wisc.edu/>)

В центре внимания участников Бекманской встречи находилась проблема Больших Данных. В отчете утверждается следующее. *Являясь сообществом, которое в течение 45 лет раздвигало границы обработки больших наборов данных, сообщество баз данных может помочь двигаться вперед миру, управляемому данными, основываясь на собственных результатах и опыте. Тем самым, у нашего сообщества имеются уникальные возможности для решения проблемы Больших Данных, огромный потенциал для революционного воздействия.*

Для реализации этого потенциала требуется обратить особое внимание на пять областей исследований: *масштабируемые инфраструктуры больших/быстро поступающих данных; преодоление разнородностей ландшафта управления данными; сквозные обработка и интерпретация данных; облачные службы; управление различными ролями людей в жизненном цикле данных.*

Мне кажется, что сейчас еще рано комментировать прогнозы, содержащиеся в этом отчете, оценивать, действительно ли исследования, касающиеся проблемы Больших Данных, сосредотачиваются в указанных пяти областях. Нужно дождаться хотя бы следующей встречи. Но следует заметить, что создание методов и средств сквозной обработки данных (от необработанных данных до

пригодных для практического использования знаний) кажется очень объемной задачей, выходящей за рамки традиционных исследовательских работ.

В связи с этим очень важной является обсуждаемая в конце отчета проблема культуры исследований. *В последние годы тревожным явлением стало возрастающее внимание к числу публикаций и счетчикам цитирования, а к не результатам исследований.* Это мешает выполнять крупные исследовательские проекты, проводить качественные конференции и т.д. Публикации и доклады на конференциях становятся не приносящим удовольствие авторам средством представить коллегам свои достижения и результаты, а всего лишь рутинной обузой.

Увы, с этим постоянно приходится сталкиваться и российским исследователям в разных научных областях. Уже выросло целое поколение исследователей, которые никогда не писали статьи и не выступали на конференциях просто потому, что у них назрела потребность сделать это. Участники Бекманской встречи не пришли к согласию о способах решения этой проблемы, но очевидно, что нужно стремиться к тому, чтобы написание и публикация статьи приносили авторам радость, а не были навязанной извне дополнительной неприятной нагрузкой.

9. Заключение

Отчеты о встречах специалистов исследовательского сообщества исследований баз данных очень полезны для всех людей, интересующихся технологиями баз данных. Они дают понять, когда и чем руководствовались исследователи в своей работе, какие на них воздействовали внешние силы, всегда ли выбираемые направления исследований соответствовали потребностям развития общей технологии.

В этой статье я попытался обеспечить ретроспективный взгляд в многолетнюю историю баз данных после первой встречи исследователей в 1988 г. в Лагуна-Бич. Мои размышления, безусловно, субъективны, и с ними совсем не обязательно нужно соглашаться, но это размышления заинтересованного человека, который все эти годы ждал новых встреч, жадно читал (и с удовольствием переводил и/или комментировал) отчеты и соизмерял свою собственную работу с мнением и прогнозами участников этих встреч.

Статья написана вдогонку к моему выступлению на Семинаре Московской секции ACM SIGMOD [66] в конце 2015 – начале 2016 г., хотя многие высказанные тогда мной соображения в статье были пересмотрены.

Список литературы

- [1]. Philip A. Bernstein, Umeshwar Dayal, David J. DeWitt et al. Future Directions in DBMS Research. ACM SIGMOD Record, vol. 18, № 1, 1989, pp. 17-26
- [2]. Сергей Кузнецов. Будущие направления исследований в области баз данных: десять лет спустя. http://citforum.ru/database/articles/future_01.shtml, дата обращения 2 апреля 2017 г.

- [3]. Abraham Silberschatz, Michael Stonebraker, and Jeffrey D. Ullman. Database Systems: Achievements and Opportunities. Communications of the ACM, vol. 34, № 10, 1991, pp. 110-120,
- [4]. Abraham Silberschatz, Michael Stonebraker, Jeffrey D. Ullman: Database Research: Achievements and Opportunities Into the 21st Century. SIGMOD Record, vol. 25, № 1, 1996, pp. 52-63. Перевод на русский язык: http://citforum.ru/database/classics/nfs_report, дата обращения 2 апреля 2017 г./
- [5]. Avi Silberschatz, Stan Zdonik et al. Strategic Directions in Database Systems – Breaking Out of the Box. ACM Computing Surveys, vol. 28, № 4, Dec 1996, pp. 764-778. Перевод на русский язык: http://citforum.ru/database/classics/nfs_report2/, дата обращения 2 апреля 2017 г.
- [6]. Philip A. Bernstein, Michael L. Brodie, Stefano Ceri et. al. The Asilomar Report on Database Research. SIGMOD Record, vol. 27, № 4, 1998, pp. 74-80. Перевод на русский язык: http://citforum.ru/database/digest/asil_01.shtml, дата обращения 2 апреля 2017 г.
- [7]. Serge Abiteboul, Rakesh Agrawal, Philip A. Bernstein et al. The Lowell Database Research Self-Assessment. Communications of the ACM, vol 48, № 5, 2005, pp. 111-118
- [8]. Сергей Кузнецов. Крупные проблемы и текущие задачи исследований в области баз данных. <http://citforum.ru/database/articles/problems/>, 2005, дата обращения 2 апреля 2017 г.
- [9]. Rakesh Agrawal, Anastasia Ailamaki, Philip A. Bernstein, et. al. The Claremont Report on Database Research. Communications of the ACM, vol. 52, № 6, 2009, pp. 56-65. Пересказ с комментариями: http://citforum.ru/database/articles/claremont_report/, дата обращения 2 апреля 2017 г.
- [10]. Daniel Abadi, Rakesh Agrawal, Anastasia Ailamaki, et al. The Beckman Report on Database Research. ACM SIGMOD Record, vol. 43, № 3, September 2014, pp. 61-70. Перевод на русский язык: http://citforum.ru/database/articles/beckman_report/, дата обращения 4 мая 2017 г.
- [11]. Keen, P. G. W. and M. S. Scott Morton. Decision support systems: an organizational perspective. Reading, Mass., Addison-Wesley Pub. Co., 1978
- [12]. Сергей Кузнецов. MapReduce: внутри, снаружи или сбоку от параллельных СУБД?. Труды ИСП, т. 19, 2010, стр. 35-40
- [13]. Edward A. Feigenbaum and Pamela McCorduck. The fifth generation: Japan's computer challenge to the world. Creative Computing Magazine, volume 10, Number 08, August 1984, pp. 103-111.
- [14]. David DeWitt, Jim Gray. Parallel database systems: the future of high performance database systems. Communications of the ACM, vol. 35, Issue 6, June 1992, pp. 85-98
- [15]. Marianne Winslett. Jim Gray speaks out. ACM SIGMOD Record, vol. 32, Issue 1, March 2003, pp. 53-61
- [16]. Gartner IT Glossary. Database Appliances. <http://www.gartner.com/it-glossary/database-appliances>, дата обращения 22.03.2017
- [17]. Exadata. <https://ru.wikipedia.org/wiki/Exadata>, дата обращения 22.03.2017
- [18]. Michael Stonebraker. My Top 10 Assertions About Data Warehouses. BLOG@CACM, August 26, 2010. Перевод на русский язык: <http://citforum.ru/gazeta/166/>, 2010 г., дата обращения 22.03.2017
- [19]. С.О. Приказчиков, П.С. Костенецкий. Применение графических ускорителей для обработки запросов над сжатыми данными в параллельных системах баз данных. Вестник Южно-Уральского государственного университета. Серия: Вычислительная математика и информатика, т. 4, вып. 1, 2015, стр. 64-70

- [20]. M. Rozier, V. Abrossimov, F. Armand, I. Boule, M. Gien, M. Guillemont, F. Herrmann, C. Kaiser, S. Langlois, P. Léonard, and W. Neuhauser. CHORUS Distributed Operating Systems. *Computing Systems*, vol. 1, No. 4, Fall 1988, pp. 305-370
- [21]. Igor Burdonov, Victor Ivannikov, German Kopytov, Alexander Kosachev, Sergei Kuznetsov. The CLOS project: Towards an object-oriented environment for application development. *Next Generation Information System Technology. Lecture Notes in Computer Science (LNCS)*, vol. 504, 1991, pp. 422-427, DOI: 10.1007/3-540-54141-1_23
- [22]. David B. Golub, Daniel P. Julin, Richard F. Rashid, Richard P. Draves, Randall W. Dean, Alessandro Forin, Joseph Barrera, Hideyuki Tokuda, Gerald Malan, David Bohman. Microkernel operating system architecture and mach. In *Proceedings of the USENIX Workshop on Micro-Kernels and Other Kernel Architectures*, 1992, pp. 11-30
- [23]. QNX Operating Systems. <http://www.qnx.com/content/qnx/en/products/neutrino-rtos/index.html>, дата обращения 22.03.2017
- [24]. Andrew Tanenbaum, Raja Appuswamy, Herbert Bos, Lorenzo Cavallaro, Cristiano Giuffrida, Tomáš Hrubý, Jorrit Herder, Erik van der Kouwe, David van Moelenbroek. MINIX 3: Status Report and Current Research. *login*, vol. 35, № 3, June 2010, pp. 7-13
- [25]. С.Д. Кузнецов. Стандарты языка реляционных баз данных SQL: краткий обзор. СУБД, № 2, 1996, стр. 6-36. http://citforum.ru/database/articles/art_2.shtml, дата обращения 22.03.2017
- [26]. The 1995 SQL Reunion: People, Projects, and Politics. Edited by Paul McJones, August 20, 1997 (2nd edition), http://www.mcjones.org/System_R/SQL_Reunion_95/SRC-1997-018.pdf, дата обращения 22.03.2017. Имеется перевод на русский язык: <http://citforum.ru/database/digest/sql1.shtml>, дата обращения 22.03.2017.
- [27]. Philip L. Frana. Oral history interview with Donald D. Chamberlin. Charles Babbage Institute, 2001. <http://hdl.handle.net/11299/107215>, дата обращения 22.03.2017
- [28]. Malcolm Atkinson, François Bancilhon, David DeWitt, Klaus Dittrich, David Maier, and Stanley Zdonik. The Object-Oriented Database System Manifesto. *Proc. 1st International Conference on Deductive and Object-Oriented Databases*, Kyoto, Japan. New York, N.Y.: Elsevier Science, 1989, pp. 223-240. Имеется перевод на русский язык: http://citforum.ru/database/classics/oo_manifesto/, дата обращения 25.03.2017
- [29]. M. Stonebraker, L. Rowe, B. Lindsay, J. Gray, M. Carey, M. Brodie, Ph. Bernstein, D. Beech. Third-Generation Data Base System Manifesto. *ACM SIGMOD Record* 19, № 3, 1990, pp. 31-44. Имеется перевод на русский язык: <http://citforum.ru/database/classics/manifest/>, дата обращения 25.03.2017
- [30]. Michael Stonebraker. The Land Sharks Are on the Squawk Box. *Communications of the ACM*, vol. 59, issue 2, 2016, pp. 74-83
- [31]. С.Д. Кузнецов. Объектно-реляционные базы данных: прошедший этап или недооцененные возможности? Труды ИСП РАН, т. 13, часть 2, 2007, стр. 115-140
- [32]. M. N. Grinev, S. D. Kuznetsov. UQL: A UML-based Query Language for Integrated Data. *Programming and Computer Software*, vol. 28, issue 4, 2002, pp 189-196. DOI: 10.1023/A:1016366916304
- [33]. M. Stonebraker, D. J. Abadi, A. Batkin, X. Chen, M. Cherniack, M. Ferreira, E. Lau, A. Lin, S. R. Madden, E. J. O'Neil, P. E. O'Neil, A. Rasin, N. Tran, and S. B. Zdonik. C-Store: A Column-Oriented DBMS. In *VLDB*, 2005, pp. 553-564
- [34]. Michael Stonebraker, Samuel Madden, Daniel J. Abadi, Stavros Harizopoulos, Nabil Hachem, Pat Helland. The End of an Architectural Era (It's Time for a Complete Rewrite). *Proceedings of VLDB*, 2007, pp. 1150-1160. Имеется перевод на русский язык: http://citforum.ru/database/articles/end_of_arch_era/, дата обращения 25.03.2017

- [35]. Vertica, <https://www.vertica.com/>, дата обращения 26.03.2017
- [36]. VoltDB, <https://www.voltdb.com/>, дата обращения 26.03.2017
- [37]. Кузнецов С.Д., Посконин А.В. Системы управления данными категории NoSQL. *Программирование*, том 40, № 6, стр. 34-47, 2014, http://www.ispras.ru/publications/2014/nosql_data_management_systems/, дата обращения 26.03.2017
- [38]. Daniel Newman. Data As A Service: The Big Opportunity For Business. <https://www.forbes.com/sites/danielnewman/2017/02/07/data-as-a-service-the-big-opportunity-for-business/#47708f3c24d9>, дата обращения 26.03.2017
- [39]. Сергей Кузнецов. Когда, как и зачем стоит применять теорему CAP? Открытые системы. СУБД, № 04, 2012, стр. 56-59, <https://www.osp.ru/os/2012/04/13015765/>, дата обращения 26.03.2017
- [40]. Jim Gray, Surajit Chaudhuri, Adam Bosworth, Andrew Layman, Don Reichart, Murali Venkatrao, Frank Pellow, Hamid Pirahesh. Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Totals. *Data Mining and Knowledge Discovery*, vol. 1, issue 1, 1997, pp 29-53
- [41]. Carl Nolan. Manipulate and Query OLAP Data Using ADOMD and Multidimensional Expressions. *Microsoft Systems Journal*, vol 14, № 8, 1999, <https://www.microsoft.com/msj/0899/mdx/mdx.aspx>, дата обращения 28.03.2017
- [42]. Oracle Database, Data Warehousing Guide, 10g Release 2 (10.2), 2005, https://docs.oracle.com/cd/B19306_01/server.102/b14223.pdf, дата обращения 28.03.2017
- [43]. Philip A. Bernstein, Umeshwar Dayal. An Overview of Repository Technology. In *VLDB '94, Proceedings of the 20th International Conference on Very Large Data Bases*, September 12 - 15, 1994, pp. 705-713
- [44]. James Gosling, Bill Joy, Guy Steele. The Java Language Specification. Addison Wesley, 1996. <http://titanium.cs.berkeley.edu/doc/java-langspec-1.0.pdf>, дата обращения 28.03.2017
- [45]. Sun Microsystems. JavaBeans API specification, version 1.01, August 1997, http://download.oracle.com/otn-pub/jcp/7224-javabeans-1.01-fr-spec-oth-JSpec/beans.101.pdf?AuthParam=1490694903_89224574569a2575b5c6d8a54a8c2ebc, дата обращения 28.03.2017
- [46]. Graham Hamilton, Rick Cattell. JDBC: A Java SQL API, Version 1.20. Sun Microsystems Inc., January 1997, <http://www.dcs.ed.ac.uk/teaching/cs2/prac6/jdbc-spec-0120.pdf>, дата обращения 29.03.2017
- [47]. Surajit Chaudhuri, Gerhard Weikum. Rethinking Database System Architecture: Towards a Self-Tuning RISC-Style Database System. In *Proceeding VLDB '00 Proceedings of the 26th International Conference on Very Large Data Bases*, September 10 - 14, 2000, pp. 1-10, <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.35.9260>, дата обращения 29.03.2017
- [48]. Michael Stonebraker, Uğur Çetintemel. «One Size Fits All»: An Idea Whose Time Has Come and Gone. In *Proceeding ICDE '05 Proceedings of the 21st International Conference on Data Engineering*, April 05 - 08, 2005, pp. 2-11. Перевод на русский язык: http://citforum.ru/database/articles/one_size_fits_all/, дата обращения 29.03.2017
- [49]. Database SQL Tuning Guide. Chapter 4, Query Optimizer Concepts. https://docs.oracle.com/database/121/TGSQL/tgsql_optncpt.htm#TGSQL192, дата обращения 29.03.2017
- [50]. Tomasz Imielinski, Heikki Mannila. A database perspective on knowledge discovery. *Communications of the ACM*, vol. 39, issue 11, 1996, pp. 58-64

- [51]. Surajit Chaudhuri, Vivek Narasayya. AutoAdmin «what-if» index analysis utility. In Proceeding SIGMOD '98 Proceedings of the 1998 ACM SIGMOD international conference on Management of data, Seattle, Washington, USA — June 01 - 04, 1998, pp. 367-378
- [52]. Nicolas Bruno, Surajit Chaudhuri, Arnd Christian Künig, Vivek Narasayya, Ravi Ramamurthy, and Manoj Syamala. AutoAdmin Project at Microsoft Research: Lessons Learned. Bulletin of the Technical Committee on Data Engineering, Vol. 34, №. 4, December 2011, pp. 12-19
- [53]. Pete Belknap, John Beresniewicz, Benoit Dageville, Karl Dias, Uri Shaft, Khaled Yagoub. A Decade of Oracle Database Manageability. Bulletin of the Technical Committee on Data Engineering, Vol. 34, №. 4, December 2011, pp. 20-27
- [54]. С.Д. Кузнецов, А.А. Прохоров. Алгоритмы управления буферным пулом СУБД при работе с флэш-накопителями. Труды ИСП РАН, том 23, 2012, стр. 173-194. DOI: 10.15514/ISPRAS-2012-23-11
- [55]. Michael Stonebraker. The Design of the POSTGRES Storage System. In Proceeding VLDB '87 Proceedings of the 13th International Conference on Very Large Data Bases, September 01 - 04, 1987, pp. 289-300
- [56]. Justin DeBrabant, Joy Arulraj, Andrew Pavlo, Michael Stonebraker, Stanley B. Zdonik, Subramanya Dulloor. A Prolegomenon on OLTP Database Systems for Non-Volatile Memory. ADMS 2014, Fifth International Workshop on Accelerating Data Management Systems Using Modern Processor and Storage Architectures, Monday, September 1, 2014, pp. 57-63
- [57]. Ismail Oukid, Wolfgang Lehner, Towards a Single-Level Database Architecture on Non-Volatile Memory (Presentation Abstract). 8th Annual Non-Volatile Memories Workshop 2017, University of California, San Diego - Price Center Ballroom East, March 12-14, 2017, <http://nvmw.ucsd.edu/2017/assets/abstracts/50>, дата обращения 31.03.2017
- [58]. Antonino Tumeo, Simone Secchi, and Oreste Villa. Designing Next-Generation Massively Multithreaded Architectures for Irregular Applications. Computer, vol. 45, № 8, 2012, pp. 53-61
- [59]. Harald Schönig. Tamino - A DBMS Designed for XML. In Proceeding ICDE '01 Proceedings of the 17th International Conference on Data Engineering, April 02 - 06, 2001, p. 149
- [60]. Andrey Fomichev, Maxim Grinev, Sergey Kuznetsov. Sedna: A Native XML DBMS. Lecture Notes in Computer Science, vol 3831, 2006, pp. 272-281. DOI: 10.1007/11611257_25
- [61]. Michael Franklin, Alon Halevy, David Maier. From Databases to Dataspace: A New Abstraction for Information Management, SIGMOD Record, Vol. 34, No. 4, Dec. 2005, pp. 27-33. Перевод на русский язык: http://citforum.ru/database/articles/from_db_to_ds/, дата обращения 2 апреля 2017 г.
- [62]. Paresh V. Virparia, Sanjay H. Buch, Roohana F. Parabia. Trade and Tricks: Traditional vs. Virtual Data Warehouse, An International Journal of Advanced Engineering & Applications, January 2010, pp.:225-239
- [63]. Bill Inmon. The Elusive Virtual Data Warehouse. <http://www.b-eye-network.com/view/9956>, March 19, 2009, дата обращения 2 апреля 2017 г.
- [64]. Vijay Gadepally, Peinan Chen, Jennie Duggan, Aaron Elmore, Brandon Haynes, Jeremy Kepner, Samuel Madden, Tim Mattson, Michael Stonebraker. The BigDAWG Polystore System and Architecture. In Proceedings 2016 IEEE High Performance Extreme Computing Conference (HPEC), pp. 1-6, <https://arxiv.org/pdf/1609.07548.pdf>, дата обращения 2 апреля 2017 г.

- [65]. Ippokratis Pandis, Ryan Johnson, Nikos Hardavellas, Anastasia Ailamaki. Proceedings of the VLDB Endowment, Vol. 3, No. 1, 2010, pp. 928-939. Перевод на русский язык: http://citforum.ru/database/articles/ailamaki_vldb2010/, дата обращения 2 апреля 2017 г.
- [66]. Ежемесячный семинар Московской Секции ACM SIGMOD. С.Д. Кузнецов. 25 лет прогнозов: что день грядущий нам готовит? Часть 1, 24 декабря 2015 г., <http://synthesis.ipi.ac.ru/sigmod/seminar/s20151224>, Часть 2, 21 января 2016 г., <http://synthesis.ipi.ac.ru/sigmod/seminar/s20160121>, дата обращения 03.04.2017

Data Management: 25 Years of Forecasts

Kuznetsov S.D. <kuzloc@ispras.ru>

*Institute for System Programming of the Russian Academy of Sciences,
25, Alexander Solzhenitsyn st., Moscow, 109004, Russia
Lomonosov Moscow State University,
GSP-1, Leninskie Gory, Moscow, 119991, Russia.
Moscow Institute of Physics and Technology (State University)
9 Institutskiy per., Dolgoprudny, Moscow Region, 141700, Russia*

Abstract. In October 2013, the eighth meeting of researchers in the field of databases was held. The first such meeting took place in February 1988, so that 25 years passed between them. After each meeting, a report was published containing an overview of the current state of the field and a research program for the nearest future, a kind of set of forecasts for the development of research activities. This paper looks at the most interesting forecasts from the reports of the research meetings, discusses how they proved to be valid, to what extent they were true or not. Among the various problems of database technology under consideration are the following: the role of specialized hardware in building effective DBMS; SQL and database applications; perspectives of object-relational extensions; distributed heterogeneous database systems; databases and Web; databases and data warehouses, OLAP and data mining; component organization of DBMS; query optimization criteria; self-tuning and self-management of DBMS; DBMS architecture and new hardware capabilities: SSD, non-volatile memory, massively multithreaded processors; specialized DBMS; data fusion and data spaces; the Big Data problem and the response to it in the database community; architectural shifts in computing.

Keywords: database research community reports; technology forecasts; data analysis; database machine; SQL; manifest of future database systems; scalability; heterogeneity; distribution; data warehouse; extendibility; query optimization; self-tuning; SSD; non-volatile memory; data space; Big Data

DOI: 10.15514/ISPRAS-2017-29(2)-5

For citation: Kuznetsov S.D. Data Management: 25 years of Forecasts. *Trudy ISP RAN/Proc.ISP RAS*, vol. 29, issue 2, 2017, pp. 117-160 (in Russian). DOI: 10.15514/ISPRAS-2017-29(2)-5

References

- [1]. Philip A. Bernstein, Umeshwar Dayal, David J. DeWitt et al. Future Directions in DBMS Research. ACM SIGMOD Record, vol. 18, № 1, 1989, pp. 17-26
- [2]. Sergey Kuznetsov. Future directions of database research: Ten years later. http://citforum.ru/database/articles/future_01.shtml (in Russian), accessed 02.04.2017
- [3]. Abraham Silberschatz, Michael Stonebraker, and Jeffrey D. Ullman. Database Systems: Achievements and Opportunities. Communications of the ACM, vol. 34, № 10, 1991, pp. 110-120
- [4]. Abraham Silberschatz, Michael Stonebraker, Jeffrey D. Ullman: Database Research: Achievements and Opportunities Into the 21st Century. SIGMOD Record, vol. 25, № 1, 1996, pp. 52-63
- [5]. Avi Silberschatz, Stan Zdonik et al. Strategic Directions in Database Systems – Breaking Out of the Box. ACM Computing Surveys, vol. 28, No. 4, Dec 1996, 764-778
- [6]. Philip A. Bernstein, Michael L. Brodie, Stefano Ceri et. al. The Asilomar Report on Database Research. SIGMOD Record, vol. 27, № 4, 1998, pp. 74-80
- [7]. Serge Abiteboul, Rakesh Agrawal, Philip A. Bernstein et. al. The Lowell Database Research Self-Assessment. Communications of the ACM, vol. 48, № 5, 2005, pp. 111-118
- [8]. Sergey Kuznetsov. Major problems and current tasks of database. <http://citforum.ru/database/articles/problems/>, 2005 (in Russian), accessed 02.04.2017
- [9]. Rakesh Agrawal, Anastasia Ailamaki, Philip A. Bernstein, et. al. The Claremont Report on Database Research. Communications of the ACM, vol. 52, No. 6, 2009, pp. 56-65
- [10]. Daniel Abadi, Rakesh Agrawal, Anastasia Ailamaki, et. al. The Beckman Report on Database Research. ACM SIGMOD Record, vol. 43, issue 3, September 2014, pp. 61-70.
- [11]. Keen, P. G. W. and M. S. Scott Morton. Decision support systems: an organizational perspective. Reading, Mass., Addison-Wesley Pub. Co., 1978
- [12]. Sergey Kuznetsov. MapReduce: inside, outside or side of parallel DBMS? Trudy ISP RAN/Proc. ISP RAS, vol. 19, 2010, pp. 35-40 (in Russian)
- [13]. Edward A. Feigenbaum and Pamela McCorduck. The fifth generation: Japan's computer challenge to the world. Creative Computing Magazine, vol. 10, No. 08, August 1984, pp. 103-111.
- [14]. David DeWitt, Jim Gray. Parallel database systems: the future of high performance database systems. Communications of the ACM, vol. 35, issue 6, June 1992, pp. 85-98
- [15]. Marianne Winslett. Jim Gray speaks out. ACM SIGMOD Record, Volume 32, Issue 1, March 2003, pp. 53-61
- [16]. Gartner IT Glossary. Database Appliances. <http://www.gartner.com/it-glossary/database-appliances>, accessed 22.03.2017
- [17]. Exadata. <https://ru.wikipedia.org/wiki/Exadata>, accessed 22.03.2017
- [18]. Michael Stonebraker. My Top 10 Assertions About Data Warehouses. BLOG@CACM, August 26, 2010
- [19]. Stepan O. Prikazchikov, Pavel S. Kostenetskiy. Using graphics accelerators for query processing over compressed data in parallel database systems. Bulletin of the South Ural State University. Series «Computational Mathematics and Software Engineering», vol. 4, issue 1, 2015, pp. 64-70 (in Russian)
- [20]. M. Rozier, V. Abrossimov, F. Armand, I. Boule, M. Gien, M. Guillemont, F. Herrmann, C. Kaiser, S. Langlois, P. Léonard, and W. Neuhauser. CHORUS Distributed Operating Systems. Computing Systems, vol. 1, No. 4, Fall 1988, pp. 305-370
- [21]. Igor Burdonov, Victor Ivannikov, German Kopytov, Alexander Kosachev, Sergei Kuznetsov. The CLOS project: Towards an object-oriented environment for application development. Next Generation Information System Technology. Lecture Notes in Computer Science (LNCS), vol. 504, 1991, pp. 422-427, DOI: 10.1007/3-540-54141-1_23
- [22]. David B. Golub, Daniel P. Julin, Richard F. Rashid, Richard P. Draves, Randall W. Dean, Alessandro Forin, Joseph Barrera, Hideyuki Tokuda, Gerald Malan, David Bohman. Microkernel operating system architecture and mach. In Proceedings of the USENIX Workshop on Micro-Kernels and Other Kernel Architectures, 1992, pp. 11-30
- [23]. QNX Operating Systems. <http://www.qnx.com/content/qnx/en/products/neutrino-rtos/index.html>, accessed 22.03.2017
- [24]. Andrew Tanenbaum, Raja Appuswamy, Herbert Bos, Lorenzo Cavallaro, Cristiano Giuffrida, Tomáš Hrubý, Jorrit Herder, Erik van der Kouwe, David van Moolenbroek. MINIX 3: Status Report and Current Research. ;login, vol. 35, № 3, June 2010, pp. 7-13
- [25]. S.D. Kuznetsov. SQL Relational Database Language Standards: An Brief Overview. DBMS, № 2, 1996, pp. 6-36 (in Russian)
- [26]. The 1995 SQL Reunion: People, Projects, and Politics. Edited by Paul McJones, August 20, 1997 (2nd edition), http://www.mcjones.org/System_R/SQL_Reunion_95/SRC-1997-018.pdf, accessed 22.03.2017
- [27]. Philip L. Frana. Oral history interview with Donald D. Chamberlin. Charles Babbage Institute, 2001. <http://hdl.handle.net/11299/107215>, accessed 22.03.2017
- [28]. Malcolm Atkinson, Francois Bancilhon, David DeWitt, Klaus Dittrich, David Maier, and Stanley Zdonik. The Object-Oriented Database System Manifesto. Proc. 1st International Conference on Deductive and Object-Oriented Databases, Kyoto, Japan. New York, N.Y.: Elsevier Science, 1989, pp. 223-240.
- [29]. M. Stonebraker, L. Rowe, B. Lindsay, J. Gray, M. Carey, M. Brodie, Ph. Bernstein, D. Beech. Third-Generation Data Base System Manifesto. ACM SIGMOD Record 19, № 3, 1990, pp. 31-44
- [30]. Michael Stonebraker. The Land Sharks Are on the Squawk Box. Communications of the ACM, vol. 59, issue 2, 2016, pp. 74-83
- [31]. S.D. Kuznetsov. Object-relational databases: past stage or undervalued capabilities? Trudy ISP RAN/Proc. ISP RAS, vol. 13, part 2, 2007, pp. 115-140 (in Russian)
- [32]. M. N. Grinev, S. D. Kuznetsov. UQL: A UML-based Query Language for Integrated Data. Programming and Computer Software, vol. 28, issue 4, 2002, pp 189–196. DOI: 10.1023/A:1016366916304
- [33]. M. Stonebraker, D. J. Abadi, A. Batkin, X. Chen, M. Cherniack, M. Ferreira, E. Lau, A. Lin, S. R. Madden, E. J. O’Neil, P. E. O’Neil, A. Rasin, N. Tran, and S. B. Zdonik. C-Store: A Column-Oriented DBMS. In VLDB, 2005, pp. 553–564
- [34]. Michael Stonebraker, Samuel Madden, Daniel J. Abadi, Stavros Harizopoulos, Nabil Hachem, Pat Helland. The End of an Architectural Era (It's Time for a Complete Rewrite). Proceedings of VLDB, 2007, pp. 1150-1160
- [35]. Vertica, <https://www.vertica.com/>, accessed 26.03.2017
- [36]. VoltDB, <https://www.voltdb.com/>, accessed 26.03.2017
- [37]. S. D. Kuznetsov, A. V. Poskonin. Programming and Computer Software. November 2014, Volume 40, Issue 6, pp 323-332. DOI: 10.1134/S0361768814060152
- [38]. Daniel Newman. Data As A Service: The Big Opportunity For Business. <https://www.forbes.com/sites/danielnewman/2017/02/07/data-as-a-service-the-big-opportunity-for-business/#47708f3c24d9>, accessed 26.03.2017

- [39]. Sergey Kuznetsov. When, How and Why the CAP Theorem Should Be Applied? Open Systems. DBMS, № 04, 2012, pp. 56-59 (in Russian)
- [40]. Jim Gray, Surajit Chaudhuri, Adam Bosworth, Andrew Layman, Don Reichart, Murali Venkatrao, Frank Pellow, Hamid Pirahesh. Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Totals. Data Mining and Knowledge Discovery, vol. 1, issue 1, 1997, pp 29–53
- [41]. Carl Nolan. Manipulate and Query OLAP Data Using ADOMD and Multidimensional Expressions. Microsoft Systems Journal, vol 14, № 8, 1999, <https://www.microsoft.com/msj/0899/mdx/mdx.aspx>, дата обращения 28.03.2017
- [42]. Oracle Database, Data Warehousing Guide, 10g Release 2 (10.2), 2005, https://docs.oracle.com/cd/B19306_01/server.102/b14223.pdf, accessed 28.03.2017
- [43]. Philip A. Bernstein, Umeshwar Dayal. An Overview of Repository Technology. In VLDB '94, Proceedings of the 20th International Conference on Very Large Data Bases, September 12 - 15, 1994, pp. 705-713
- [44]. James Gosling, Bill Joy, Guy Steele. The Java Language Specification. Addison Wesley, 1996. <http://titanium.cs.berkeley.edu/doc/java-langs-spec-1.0.pdf>, accessed 28.03.2017
- [45]. Sun Microsystems. JavaBeans API specification, version 1.01, August 1997, http://download.oracle.com/otn-pub/jcp/7224-javabeans-1.01-fr-spec-oth-JSpec/beans.101.pdf?AuthParam=1490694903_89224574569a2575b5c6d8a54a8c2ebc, accessed 28.03.2017
- [46]. Graham Hamilton, Rick Cattell. JDBC: A Java SQL API, Version 1.20. Sun Microsystems Inc., January 1997, <http://www.dcs.ed.ac.uk/teaching/cs2/prac6/jdbc-spec-0120.pdf>, accessed 29.03.2017
- [47]. Surajit Chaudhuri, Gerhard Weikum. Rethinking Database System Architecture: Towards a Self-Tuning RISC-Style Database System. In Proceeding VLDB '00 Proceedings of the 26th International Conference on Very Large Data Bases, September 10 - 14, 2000, pp. 1-10
- [48]. Michael Stonebraker, Uğur Çetintemel. «One Size Fits All»: An Idea Whose Time Has Come and Gone. In Proceeding ICDE '05 Proceedings of the 21st International Conference on Data Engineering, April 05 - 08, 2005, pp. 2-11
- [49]. Database SQL Tuning Guide. Chapter 4, Query Optimizer Concepts. https://docs.oracle.com/database/121/TGSQL/tgsql_optcnpt.htm#TGSQL192, accessed 29.03.2017
- [50]. Tomasz Imielinski, Heikki Mannila. A database perspective on knowledge discovery. Communications of the ACM, vol. 39, issue 11, 1996, pp. 58-64
- [51]. Surajit Chaudhuri, Vivek Narasayya. AutoAdmin «what-if» index analysis utility. In Proceeding SIGMOD '98 Proceedings of the 1998 ACM SIGMOD international conference on Management of data, Seattle, Washington, USA — June 01 - 04, 1998, pp. 367-378
- [52]. Nicolas Bruno, Surajit Chaudhuri, Arnd Christian Künig, Vivek Narasayya, Ravi Ramamurthy, and Manoj Syamala. AutoAdmin Project at Microsoft Research: Lessons Learned. Bulletin of the Technical Committee on Data Engineering, Vol. 34, №. 4, December 2011, pp. 12-19
- [53]. Pete Belknap, John Beresiewicz, Benoit Dageville, Karl Dias, Uri Shaft, Khaled Yagoub. A Decade of Oracle Database Manageability. Bulletin of the Technical Committee on Data Engineering, Vol. 34, №. 4, December 2011, pp. 20-27
- [54]. Kuznetsov S.D., Prokhorov A.A. Flash-based algorithms of database buffer management. Trudy ISP RAN/Proc. ISP RAS, vol. 23, 2012, pp. 173-194 (in Russian). DOI: 10.15514/ISPRAS-2012-23-11

- [55]. Michael Stonebraker. The Design of the POSTGRES Storage System. In Proceeding VLDB '87 Proceedings of the 13th International Conference on Very Large Data Bases, September 01 - 04, 1987, pp. 289-300
- [56]. Justin DeBrabant, Joy Arulraj, Andrew Pavlo, Michael Stonebraker, Stanley B. Zdonik, Subramanya Dullloor. A Prolegomenon on OLTP Database Systems for Non-Volatile Memory. ADMS 2014, Fifth International Workshop on Accelerating Data Management Systems Using Modern Processor and Storage Architectures, Monday, September 1, 2014, pp. 57-63
- [57]. Ismail Oukid, Wolfgang Lehner, Towards a Single-Level Database Architecture on Non-Volatile Memory (Presentation Abstract). 8th Annual Non-Volatile Memories Workshop 2017, University of California, San Diego - Price Center Ballroom East, March 12-14, 2017, <http://nvmw.ucsd.edu/2017/assets/abstracts/50>, accessed 31.03.2017
- [58]. Antonino Tumeo, Simone Secchi, and Oreste Villa. Designing Next-Generation Massively Multithreaded Architectures for Irregular Applications. Computer, vol. 45, № 8, 2012, pp. 53-61
- [59]. Harald Schöning. Tamino - A DBMS Designed for XML. In Proceeding ICDE '01 Proceedings of the 17th International Conference on Data Engineering, April 02 - 06, 2001, p. 149
- [60]. Andrey Fomichev, Maxim Grinev, Sergey Kuznetsov. Sedna: A Native XML DBMS. Lecture Notes in Computer Science, vol 3831, 2006, pp. 272-281. DOI: 10.1007/11611257_25
- [61]. Michael Franklin, Alon Halevy, David Maier. From Databases to Dataspaces: A New Abstraction for Information Management, SIGMOD Record, Vol. 34, No. 4, Dec. 2005, pp. 27-33
- [62]. Paresv V. Virparia, Sanjay H. Buch, Roohana F. Parabia. Trade and Tricks: Traditional vs. Virtual Data Warehouse, An International Journal of Advanced Engineering & Applications, January 2010, pp.:225-239
- [63]. Bill Inmon. The Elusive Virtual Data Warehouse. <http://www.b-eye-network.com/view/9956>, March 19, 2009, accessed 2 апреля 2017 г.
- [64]. Vijay Gadepally, Peinan Chen, Jennie Duggan, Aaron Elmore, Brandon Haynes, Jeremy Kepner, Samuel Madden, Tim Mattson, Michael Stonebraker. The BigDAWG Polystore System and Architecture. In Proceedings 2016 IEEE High Performance Extreme Computing Conference (HPEC), pp. 1-6, <https://arxiv.org/pdf/1609.07548.pdf>, дата обращения 2 апреля 2017 г.
- [65]. Ippokratis Pandis, Ryan Johnson, Nikos Hardavellas, Anastasia Ailamaki. Proceedings of the VLDB Endowment, Vol. 3, No. 1, 2010, pp. 928-939
- [66]. Monthly Seminar of the Moscow Section of ACM SIGMOD. 25 years of Forecasts: What is the future preparing for us? Part 1, December 24th, 2015, <http://synthesis.ipi.ac.ru/sigmod/seminar/s20151224>, Part 2, January 21st, 2016, <http://synthesis.ipi.ac.ru/sigmod/seminar/s20160121>