第 1 届超大数据库会议（XLDB2007）大会报告（中文版）

REPORT FROM THE FIRST WORKSHOP ON EXTREMELY LARGE DATABASES

***J Becla*[*1] *and K-T Lim*[2]**

Stanford Linear Accelerator Center, Menlo Park, CA 94025, USA

[*1] *Email: becla@slac.stanford.edu*

[2] *Email: ktl@slac.stanford.edu*

温馨提示：本文由厦门大学计算机系林子雨老师翻译自 XLDB 会议网站的英文报告，转载请注明出处，仅用于学习交流，请勿用于商业用途。

[本文翻译的原始出处：厦门大学计算机系数据库实验室网站林子雨老师的超大数据库技术资料专区 http://dblab.xmu.edu.cn/XLDB]

翻译者林子雨个人主页：http://www.cs.xmu.edu.cn/linziyu

【摘要】近些年，产业界和科学界的数据集，无论在数量上还是在复杂性上，都有了巨大的增长。最大的事务型数据库和数据仓库，再也无法使用现成的商业数据库管理系统产品进行有效地存储和管理。目前，也存在其他一些专门讨论数据库和数据仓库的论坛，但是，他们通常只关注小规模数据问题，而且往往不怎么关注实际解决方案以及对数据库厂商的影响。鉴于目前的超大规模数据库的用户还比较少（但是影响力很大，并且正在迅速增加），并且缺少在超大规模数据库方面交流经验知识的机会，因此，我们组织举办了超大数据库会议。本文是这次大会的讨论和相关活动的总结报告。

【关键词】数据库;超大数据库;XLDB

1 大会总结

本次大会提供了一个交流超大规模数据库问题的论坛。与会者涵盖了一大类群体，包括科学领域和产业界的数据库密集型应用群体，DBMS（数据库管理系统）厂商和学术界群体。本次讨论会所涉及的大多数系统的数据规模，处于几百 TB 到几十 PB 之间。实际上，由于可扩展性的限制和高昂的存储处理代价，一些具有潜在价值的数据都被丢弃了。目前的情况是，产业界的数据仓库，在数据量上已经超越了科学领域。

在使用超大规模数据库方面，我们还是可以观察到科学界和产业界二者之间存在的许多共性。这些共性包括，需要进行模式发现、多维数据聚合，无法预测的查询负载，用来表达复杂分析的编程语言。二者主要的区别包括，可用性要求（产业界要求很高的可用性）、数据分布复杂性（在科学界会显得更加复杂，因为科学界包含了大量合作）、项目生命周期（在科学界，项目时间跨度可能是十几年，而在产业界则是几个季度）、使用压缩（产业界使用数据压缩，科学界不用数据压缩）。产业界和科学界这两个群体，目前都开始转向基于商业硬件集群的、并行、非共享架构，这其中，Map/Reduce 模型是一种领先的处理模型。总体而言，与会者认为，产业界和科学界在数据密集性方面不断增长，由此也不断突破数据库的处理极限。现在，产业界在数据规模上处理领先地位，而科学界在数据分析的复杂性方面处

理领先地位。

大会讨论了一些非技术方面的障碍，包括资金问题和缺少沟通的问题（包括厂商和用户之间的沟通、科学界内部之间的沟通、学术界和科学界之间的沟通）。科学领域的计算一直缺乏资金的支持，科学界群体也正在和产业界群体一样，都在努力解决数据规模和复杂性的问题，但是，前者的团队相对较小。数据库研究也缺乏资金支持。RDBMS 厂商投入了大量资金，研究可以支持几个 PB 的可扩展性解决方案，但是，至今仍然没有产生有价值的成果。科学界通常倾向于重构软件，而不是重用软件，至今也没有产生一个公共需求集合。与会者认为，在超大规模数据库领域，一旦资金和交流沟通问题得到部分解决，学术界、产业界、科学界和厂商之间就可以共同开展工作。

大会还讨论了大规模数据库系统的发展趋势和对未来的预期。用户所要求的系统的规模与数据库厂商能够有效支持的规模，这二者之间的差距不断扩大。超大数据库用户正在尝试一些新的解决方案，它们把轻量级、灵活的、专业化的组件和开放式接口融合在一起，这些开放式接口可以很容易地和廉价的商业硬件进行匹配。现有的庞大的 RDBMS（关系数据库管理系统）正面临着朝着上面这个方向进行重新设计。结构和非结构化数据并存，教科书中的方法已经无法满足要求，因为它要求完美的模式和干净的数据。在许多地方都很流行的 map/reduce 模型，缺乏高效的连接算法，因此，不大可能成为终极解决方案。最新的硬件技术，也会打算数据库技术的发展步伐，尤其是 CPU 和 I/O 能力之间的鸿沟越来越大，以及日益成熟的固态技术。

大会还讨论了未来的工作。与会者认为，合作将会带来收益。后面应该继续举办这种会议，或许应该建立一个小规模的工作组。与会者也强烈建议，建立一个专门针对数据密集型查询的、标准的测试基准，以及共享一些基础设施，比如测试环境和发布信息的 wiki。

2 关于大会

XLDB 大会为讨论超大数据库相关问题提供了一个论坛。大会于 2007 年 10 月 25 日在 SLAC(斯坦福直线加速器中心)举行，大会的主要目标是：

- 确定构建超大规模数据库的技术趋势和主要技术障碍；
- 为构建超大规模数据库的用户和数据库厂商之间搭建起沟通的桥梁；
- 了解开源项目 LSST 数据库在未来几年可以为上述两个目标做出哪些贡献。

大会的网站是：http://www-conf.slac.stanford.edu/xldb07.

附录 A 中给出了大会日程。

大会组织委员会成员包括：Jacek Becla (chair), Kian-Tat Lim, Andrew Hanushevsky 和 Richard Mount.

2.1 参会情况

大会采用邀请参会的形式，从而保证参会人员控制在尽可能小的规模，保证不用麦克风就可以达到良好的互动讨论效果。在参会人数构成上，考虑了不同群体之间的代表数量的均衡。从与会者的反馈情况和大会的效果来看，这种策略看来是很成功的。

参会的 55 个代表来自各个不同群体：产业界（数据库用户和厂商）、科学界群体（数据库用户）和学术界（数据库研究人员）。在主题讨论中，来自业界的 XLDB 用户群体主要来自 AOL, AT&T, EBay, Google 和 Yahoo!等公司。出席大会的数据库厂商包括 Greenplum, IBM, Microsoft, MySQL, Netezza, Objectivity, Oracle, Teradata 和 Vertica。来自学术界的代表包括来

自 University of Wisconsin 的 David DeWitt 教授和 来自 M.I.T.的 Michael Stonebraker 教授。来自科学界的代表包括：CERN, the Institute for Astronomy at the University of Hawaii,IPAC, George Mason University, JHU, LLNL, LSST Corp., NCSA, ORNL, PNL, SDSS, SLAC, U.C. Davis 和 U.C. Santa Cruz.

在参会用户群体的选择上，考虑了要涵盖很大范围内的数据库应用。产业界代表，范围涵盖了搜索引擎（Google 和 Yahoo!）、网络门户(AOL)、在线拍卖系统(EBay)和电信(AT&T)。科学界的代表群体，涵盖了高能物理(LHC,BaBar)、天文（(SDSS,PanSTARRS, LSST, 2MASS）以及复杂生物系统。附录 B 中给出了所有参会者的名字和单位。

2.2 结构

大会只安排了一天的日程，从而方便参会，也有助于建设性的讨论。大会的大部分时间都用于进行高度互动的讨论，当然，大会也包括了两个来自最大的科学项目的报告，其中，LHC 代表了当前的用途和高能物理，LSST 代表了未来的用途和天文。会议日程分成三个部分：

- 用户专题讨论：来自科学界和产业界的群体参与讨论，揭示超大数据库的实际用途，如何实现以及用户如何才能喜欢使用。
- 数据库厂商和学术界群体对问题的反应；
- 讨论未来的可能工作。

2.3 关于这个报告

本报告的结构，没有和大会讨论的主题保持完全一致。因为我们想要提炼出大会的主要讨论线索。

第 3 节讨论了超大数据库如何在实际生产中得到应用，并且讨论了当前的技术解决方案和问题。第 4 节 XLDB 相关群体之间的合作。第 5 节总结了与会者对 XLDB 大会的未来发展的预期。第 6 节给出了大会讨论通过的未来工作步骤。

我们已经故意弱化了特定项目的名字和与会者的名字，从而可以更好地发现科学界和产业界群体之间的共性和差别。

3 目前的解决方案

本节阐述超大数据库的当前状态和其在科学界和产业界的实践，这些都是本次大会讨论得到的结果。

3.1 规模

大会旨在讨论超大规模数据库。讨论中涉及到的大多数系统，采用的数据库的数据量都在 100TB 左右，其中，有大约 20%的科学系统的数据量超过 1PB。所有的产业界的代表所使用的数据库系统的数据量加在一起超过 10PB，其中，每个最大的系统的数据量都超过 1PB。

但是，数据量并不仅仅只用字节来描述。产业系统所使用的单个数据表的大小，都已经超过了一万亿行。在科学领域，数据库的规模也达到了几百 TB。在未来不到十年时间内，将会需要包含几万亿行的表。

数据生成的峰值速率是每小时 10 亿行，一天生成上百亿行数据是很普遍的。

所有的用户都说，虽然他们的数据库的数据量正在迅速增加，但是，只要还支付得起费

用，他们还是会继续增加更多的数据。预计未来的数据量将会是现在的 10 到 100 倍。与会者广泛认同一点，即“现在没有一个数据库厂商可以满足我们的数据库需求”。

3.2 用途

大会讨论到的最大的数据库，是用于分析的传统数据仓库的变种。它们的共同的特性包括：“一次写多次读”模型；不需要事务；支持并发负载和查询；提供对实时数据的快速访问。这些分析型系统通常和操作性系统 OLTP 是分离的，由 OLTP 系统负责生成数据。

来自科学界和产业界的代表，都谈到了高度不可预测的查询负载，超过 90%的查询都是全新的。一个短语可以很好地描述这一点，即“面向未知查询的设计”。但是，这里也会存在共性。大部分负载包含了汇总查询，需要跨越多个数据库。包含值或区间谓词的多维查询是非常普遍的，它们通常针对大量属性的不同子集进行查询。虽然，科学领域已经开发出了特定的索引方法，但是，仍然缺少可以支持这些查询的通用的索引方法，因此，仍然需要频繁地进行全表扫描。为了应付这种负载的变化，强烈建议一个设计良好的系统能够对自己的用途进行跟踪，并且能够适应最新的负载。

正像上面提到的那样，对于一些项目而言，并非所有的数据都要存入数据库。有一些数据会被丢弃掉，有些不常用的数据以及不需要诸如事务特性的数据，会在其他系统中进行管理，这些系统和文件系统比较类似，通常比较廉价，可扩展性也更好。许多科学界的用户和一部分产业界的用户会在数据库中存储汇总数据，而把细节数据存储在数据库的外面。

当然，数据库还有其他用途。面向关键事务处理的数据管理，一般都是由数据库来处理的，通常都是使用现成的 RDBMS 产品。大会还讨论了操作型数据存储，它们需要很低的访问延迟，很高的可用性，但是，具有严格定义的查询集合。上述这些其他用途，通常都不是很大的数据库，虽然它们可能用来存储最大的数据库的元数据。

这些数据库用途，通常需要使用数据库包，每个应用领域都有特定的软件包。例如，一个现成的 RDBMS 可以为一个定制的 map/reduce 系统充当数据源。一个现成的 RDBMS 可能用于保存那些来自不同系统的快速访问汇总。

3.3 硬件和压缩

为了获得处理超大规模数据所必须的可扩展性，就必须采用并行的方式。大家通常认为，I/O 吞吐量比 CPU 处理能力更加重要，即使对于科学界和产业界而言，都有一些复杂的需要大量 CPU 的处理任务。许多系统都采用了非共享的体系架构，以水平的方式对数据进行划分。在单个集群中并行运行的节点的数量，对于产业界而言，可能达到上万个，对于科学界而言，则可以达到几千个。

增加集群中节点的数量，会极大增加系统失效的概率。事实上，在大型系统中，硬件失效是一件常见的情况。当前解决系统失效的主流解决方案是，采用软件的方式解决硬件失效，而不是依赖高端的硬件来提供高可用性。一旦基于软件的、透明的失效恢复程序投入使用，大多数轻量级的系统失效（一个报告显示，对于硬盘而言，一年中失效的概率在 4-7%），就不会影响到系统的整体可用性。因此，科学界和产业界的许多项目，就可以使用很多低端的商业硬件，而不是传统的共享内存的服务器。在使用磁盘时，大都使用本地磁盘，而不是使用存储区域网络（SAN）。磁盘驱动器转轴的数量，要比磁盘总的存储空间更加重要，对于随机访问的系统而言，更是如此。Map/reduce 系统的一个优点就是，减少了随机访问。
这些大型系统通常也需要大量的电能、冷却和摆放空间。系统中的数据库组件被认为消耗了大量的上述资源，因为，磁盘阵列会产生大量的热量。

保护磁盘空间和 I/O 带宽的一种方式是压缩数据。在产业界，每个人都会以不同的方式压缩数据。但是，对于科学项目而言，通常不会压缩数据，因为数据的结构和构成都无法让数据的压缩后带来的收益大于 CPU 开销。

3.4 SQL 和关系模型

由于许多年以前的一次“大辩论”，关系模型和 SQL 查询语言开始变得非常流行，取得了巨大的成功。保持数据和处理模型的简洁性，使得关系数据库可以在各个不同领域得到很好的应用。但是，也有许多用户对于现成的关系数据库产品中存在的许多束缚条件感到不满意。

在产业界，数据通常是在高度规范化的 OLTP 系统中产生的，然后，把数据抽取到非规范化甚至是半结构化的系统中进行分析。对于科学数据而言，就很少被存储到规范化的系统中。通常情况下，几十亿到上百亿行的连接操作，性能很差，为了在分析工作中获得好的性能，用户就需要提前对数据进行连接操作，或者使用 map/reduce 模型来处理简单的、分布的连接操作。

正如上面提到的那样，现在对全表进行扫描还是比较常见的，很少会优先考虑索引，当然，数据分区在本质上也可以算作是第一个层次的索引。在这些大型系统中，列式数据库正在扮演越来越重要的角色，因为，对于典型的查询而言，它们可以显著降低 I/O 开销。

科学界和产业界都需要编程语言来处理和分析数据。比如，在产业界，会使用高级别的语言，比如 Sawzall,Pig 或 Ab Initio ETL 工具。在科学界，会使用较低级别的语言，比如 C++。当产业界确实需要 SQL 进行分析时，通常都是使用工具生成查询，而不是手工编写 SQL 代码。据一个报告显示，90%的查询都是通过工具生成的。但是，在科学界，手工编写代码更加常见，会频繁地使用较低级别的编程方式来访问数据，因为，科学界工作人员已经尝试了很多查询生成工具，但是，效果都不好。

在当前的发展阶段，对于超大数据库而言，尤其是对于数据仓库类型的超大数据库而言，对象-关系适配器，或者面向对象的数据库，也被认为是不合适的。

把科学领域的数据映射到关系模型中，针对科学领域数据开发新的处理模型，需要克服一些障碍，这对于在构建超大规模数据库的过程中实现硬件和人员的可扩展性而言是非常必要的。

3.5 操作

产业系统通常需要高的可用性，即使在高负载和备份期间，也是如此。这些系统通常会被集成到商务处理过程中。另一方面，在科学界，当一些长时间运行的、复杂的查询被中断掉，可能是无法被用户所接受的，但是，除此以外，其他情况下发生中断，哪怕是只能保证 98%的时间里系统可用，用户都可以忍受。

作为一个大型系统中的一个组件的数据库，通常不仅仅需要高可用性，它们还需要提供实时和准实时的响应。例如，在科学领域，必须实时捕捉探测器的输出，在产业界，一些快速反馈会带来直接的收入。一个产业界的项目，每天要处理多达 25TB 的数据流。另一方面，一些科学数据库，可能一年才发布一次，然后，采用最新的算法对数据进行处理，可以保证很高的质量。

这些系统的可管理性是非常重要的。没有哪个机构可以承担起大量数据库管理员的开销，因为，管理员的数量会随着系统容量的增加而增加。

在多个地方（可能是世界范围内）进行数据的复制和系统的分布，对于维护系统的可用性和性能而言，都是很有必要的，但是，这也增加了头痛的管理难题。科学界尤其受到这个问题的困扰，因为，他们的工作通常是在全球范围内几百个互不相识的人之间合作开展的，

这些人使用截然不同的硬件、软件、网络和文化环境，每个人都具备不同的经验。与此相反，产业界则可以非常严格有效地控制和管理这些配置。

3.6 软件

虽然产业界比科学界拥有更多的资源，但是，只愿意为数据库解决方案提供仅能满足需求的财力支持。两个群体通常都使用一些免费的或开源的软件，比如 Linux、MySQL 和 PostgreSQL，从而大幅降低开销。两个群体也会编写定制的软件。产业界通常倾向于实现定制的、可扩展的基础架构，包括 map/reduce 框架或列式数据库，这些都为编程人员和分析人员提供了较好的抽象架构。科学界则倾向于执行定制的、从上到下的分析，使用较低层次的数据访问层，数据访问层可以把底层的存储与应用分离开来，从而可以使得应用可以部署到各种异构的环境中。

通常来说，产业界都是处在快速变化之中，都是以季度或月为基础来运营项目；而科学界的变化节奏很慢，一个项目可能会持续很多年。实际上，产业界需要把基础设施开发工作分摊到多个年份中，科学界也需要在项目执行过程中不断更新和替换相关的技术。在这两种情况下，软件都是随着新的需求和特性的变化而不断演化的。

所有的与会者都表达了这样一种需求，即在数据库中的数据上执行大规模的计算，而不只是检索数据。从大量的数据中提取模式，以及在数据集中发现异常，是主要的应用需求。这些面向分析和发现的任务，需要交互式响应，因为，需要对假设进行不断反复地测试、细化和检验。现在，从科学界的数据生成有用的属性的算法，通常都需要昂贵的计算代价，尤其是对于浮点运算而言，代价可能比产业界多几个数量级。正因为如此，科学界通常倾向于比产业界做更多的属性预计算和汇总操作，虽然，这会降低分析和发现过程的速度。

Map/reduce 模型由于其处理模型的简单性、良好的可扩展性和容错性，已经获得了很好的认可。这个模型可以很好应用于前面讲过的全表扫描。但是，也有研究指出，这个模型的连接能力很有限，而目前的应用中，常常主要用来进行大规模数据的排序和合并。

3.7 结论

在使用超大规模数据库方面，产业界和科学界之间存在着大量的共性。科学界经常会产生大量的数据，当前，在许多不同科学领域的研究，数据都正变得越来越密集，因此，迫切需要对这些数据进行处理、搜索和分析。由此，数据库也正扮演着越来越重要的角色。产业数据仓库，在数据量方面，已经超越了科学领域，前者大约是后者的 10 倍。

产业界和科学界所使用的查询的类型，也表现出了相似性，在两个群体中，多维数据汇总和模式发现，已经变得很普遍。但是，商业分析的总体复杂性还是不如科学领域。

关系数据模型对于组织这些超大规模数据而言，仍然是有用的。产业界正在不断对关系数据库进行扩展，科学界也正努力把复杂的数据存储到关系数据库中。

两个群体都开始采用并行、非共享体系架构，采用大的商业服务器构成的集群，以及把 map/reduce 作为主要的数据处理模型。

4 合作问题

本节阐述在构建超大数据库过程中面临的非技术障碍和问题，包括社交和资金问题，这些问题在本次大会讨论中都有论及。针对某些情形，大会讨论还给出了可能的解决方案。

4.1 厂商和用户缺乏沟通

科学界和产业界使用到的数据库的规模的增加速度，已经超过了最好的数据库厂商的能力范围。目前，数据库厂商已经和一些大的用户开展合作，从而学习如何把当前的数据库技术应用到超大规模数据上，并发现未来的需求，但是，数据库厂商的前进步伐仍然无法令人满意。给人的总体感觉是，他们可能构建在落后的指标上，而不是先进的指标，这就导致他们给出的解决方案往往是针对昨天的问题，而不是当前的新问题。用户的感受是“现有的数据库产品，可以很好地解决我们五年前遭遇到的问题，但是，面对当前的新问题，却无能为力”。针对这个问题，一个可能的解释是，数据库厂商对现实世界的真实需求不是很了解，尤其是不了解大规模数据问题。甚至有与会者建议，数据库厂商代表应该转换角色，站在用户的立场上，感受一下用户的操作。

4.2 科学界内部缺乏沟通

科学界外部对科学界群体的感受是，科学界群体在软件开发层面效率很低——科学界通常重构软件，而不是重用软件。针对效率低下的问题，一个合理的解释是，研究生提供了廉价劳动力，无代价使用这些劳动力，意味着共享代码不会产生任何收益。另一方面，很大一部分科学软件无法得到重用，即使在一个科学群体内部也是无法重用，因为，这些软件都是执行一些高度专业的计算任务，这些计算任务和实验硬件紧密相关。

大的科学项目的生命周期，通常都会持续几十年，这就迫使科学家比如引入额外的层，把不同的组件隔离开来，便于执行一些无法避免的迁移操作，但是，这样做又增加了系统的复杂性。不幸的是，这些额外增加的层，一般都用于对存储模型进行抽象，而没有用于对处理模型的抽象。

总的来说，科学界群体需要进行更加努力的尝试，从而在公共需求上达成共识，编写更多高效的软件，构建更多可共享的基础设施。

4.3 学术界和科学界缺乏沟通

在过去，在数据库领域工作的计算机科学家，已经努力和其他科学家进行合作。但是，这些努力都失败了。技术性失败因素包括在数据库中支持数组和不确定性的困难性。社交方面的失败因素包括两个群体之间期望不同，计算机科学家希望产生原型系统，但是，科学家则希望一个生产型系统。这些都导致了缺乏采用最新的技术，从而无法获得这些新技术的应用效果的反馈。Jim Gray 作为一个卓越的人才，经常受到表扬，他努力尝试为两个群体搭建沟通的桥梁，这主要得益于他可以充分利用 Microsoft 的资源。

有人建议在科学实验室中引入计算机专业的研究生一起工作，但是，这个建议被认为不可行。科学项目的时间跨度很大，因此，学生无法在实验室学习期间发表能够增加自己就业砝码的论文。

让科学界和学术界再次联姻是可能的。两个群体都必须愿意开展合作，并且设定合理的预期目标。科学界必须认真重视数据库和数据管理工作，而不是把这些看作是次要附属。最迫切的需求是，科学界首先应该确定一个精炼的需求集合，比如期望的数据类型和操作类型。科学项目应该能够把一些关键数据存放到数据中心，学术界人员能够访问这些数据，并以这些数据为基础进行实验。

4.4 资金问题

高端的商业系统，购买价格昂贵，运行费用也很高。科学界当然无法为这种昂贵的商业系统支付费用，即使产业界在面对这些昂贵的价格标签时，有时候也会畏而却步。产业界的

做法是，投资构建定制的数据库系统，这种系统更加高效，性价比也高，即使系统的开发费用只能在群体内部进行分摊。但是，科学界在软件系统开发方面的投资明显不足。科学界面对和产业界一样的数据规模和复杂性问题，但是，科学界只有更小的团队来处理这些问题。产业界的前进步伐很快，这样可以缩短开发周期，因此也缩短了投资回报周期。

在计算机科学领域内部的数据库研究，也被认为投资不足。结果就是，在数据库领域，自从引入关系数据模型以后，就再也没有发生明显的技术变革。一个简洁有力的评论说：“二十年的研究，我们得到了什么，得到了 map/reduce”。

4.5 结论

对于学术界、科学界和数据库厂商群体而言，在超大规模数据方面开展合作具有很大的潜力。过去限制数据库发展的难题，现在必须被克服掉，必须加大对数据库研究和科学基础设施的投入。

5 XLDB 的未来

本节阐述数据库系统的发展趋势和未来期望。这个问题是本次大会好几个分组讨论（尤其是数据库厂商和学术界）的主题。

5.1 数据库市场的状态

本次大会讨论认为，标准的 RDBMS 技术已经无法满足超大数据库用户的需求。知名的数据库厂商没有对这个问题做出快速响应，他们没有对自己的产品进行拓展以满足超大规模数据的需求。种种迹象显示，系统能够有效支持处理的数据量和用户的需求之间的鸿沟，正变得越来越大。

与此同时，开源的通用 RDBMS 软件，在性能方面变得越来越强大，价格也越来越低。其中一些系统还设计了开放式接口，允许用户把定制的组件（比如存储引擎）接入到一个标准的、经过良好测试的框架中，从而可以让产品满足用户特定的需求。但是，开源数据库群体仍然没有解决数据库规模的问题，对于那些愿意投资开发定制软件的用户而言，他们发现这些开源数据库产品可以作为更大规模系统的一个组件。

同时，一些专业化的引擎，比如面向对象数据库、列式数据库和其他支持密集查询的 OLTP 数据库，也朝着大规模方向迈进。数据压缩（可以让 I/O 吞吐量获得数量级的提升）、有效的数据聚类、轻松的扩展能力，以及由此导致的性价比的提升，使得部署上述这些系统具有较高的价值，尽管它们之间还存在着互操作性的难题。

由于这个趋势，传统的 RDBMS 厂商，正面临着不断增加的竞争。一个与会者甚至激进地认为，在未来的十到二十年的时间里，传统的 RDBMS 厂商将会逐渐退出舞台。即使这种观点最终不会成为现实，但是，RDBMS 厂商不得不经历大量的技术革新，从而可以支持超大规模数据库。

管理大量数据集的人员，一般很不喜欢单机系统。这种系统缺乏灵活性，很难扩展和调试错误，并且会把用户捆绑到某种类型的硬件上，这种硬件通常是高端的，价格昂贵。目前的趋势是，从专业的、轻量级的组件开始构建数据管理系统，然后把这些组件和一些低端的商业硬件（CPU、内存、闪存、快速磁盘和慢速磁盘）进行结合，从而获得一个较好的平衡点。

当前，结构化数据和非结构化数据并存。教科书中的方法，需要假定一个完美的模式和干净的数据，这是不行的。今天针对大规模数据的分析，必须能够处理灵活的模式和能够生

成近似结果的不确定性数据。

Map/reduce 模型，由于其简洁性，获得了很好的认可。但是，它仍然不是终极解决方案，因为，它除了执行合并和排序以外，仍然缺少连接算法，这是一个很大的缺陷，会制约它的发展。

最后，大家也观察到，学术界的计算机科学家，不再关注核心的数据库技术。在这个领域，数据集成已经成为一个更加热门的研究问题。

5.2 硬件趋势的影响

CPU 内核的数量和处理能力正在迅速增加。仅就这点而言，数据库就必须具备大规模并行处理的能力，从而可以很好地利用 CPU 能力。这种并行能力，必须发生在所有的软件层面，包括查询执行和低级别的内部处理。

我们什么时候可以看到“光计算机”，目前还很不确定。这种技术，一旦出现，对于数据库而言，具有很大的破坏性。

磁盘正变得越来越大，越来越密集。原始的磁盘传输速率，在过去的这些年里已经得到很大的改进，但是 I/O 事务的速率，受到磁头的移动速率的制约，仍然没有明显提升。磁盘正逐渐发展成为顺序读写设备。这严重影响了数据库，因为，数据库通常是对小块数据进行随机读写。

与会者同时强调，电能和制冷通常会被忽视。在一个完整的系统中，数据库通常是一个耗电大户，因为，需要大量的磁盘转动，这会产生大量的热量。现在看起来，在未来的时间里，磁盘被固态盘取代已经成为定局。一个最有潜力的替代者就是闪存。大规模部署闪存（具有很好的随机访问能力），将会彻底改变数据集的管理方式，将会在很大程度上影响数据的分区、索引、复制和聚合。

5.3 结论

未来的十年会很有趣。在软件和硬件方面的发展趋势，将可以支持构建更大规模的数据库。学术界群体和数据库厂商的大量研究，也会大大简化构建这些系统所需要做的工作。未来的发展趋势还包括，大规模并行商业“盒子”和固态存储，以及软件行业的演化（朝着轻量级组件和专业化分析引擎方向发展）。

6 下一步工作

与会者广泛认同一点，这个大会应该在未来继续举办，本次大会应该成为未来长期合作的一个开始。一些与会者对于自己缺少空闲时间赶到苦恼，但是，大家普遍认为：“如果你不能把一些时间用于合作上面，那么，超大数据库就不是你的核心问题。”与会者认为，我们应该：

（1） 举办另一次会议；
（2） 尽快建立小的工作组；
（3） 尽快定义一个面向数据密集型查询的测试基准；
（4） 建立共享的基础设施，包括测试环境和用于发布信息的 wiki（可以发布一些对其他用户有帮助的实践经验）；
（5） 通过撰写一些观点论文，在维基百科中创建一个条目以及其他方式，来增加大家对超大数据库的认识。

6.1 下一次大会

与会者认为，下一次大会应该和本次大会类似。下一次大会应该和本次大会间隔一年左右时间再召开，从而让大家有时间在某些方面取得进展，比如建立一个较小的工作组，让一些项目积累经验，这些项目包括：LHC、PanSTARRS 以及 Google/IBM/学术界的集群。

大会的日程应该延长到 2 至 3 天，从而有更多的时间来分享经验和开展讨论。既然目前已经确定了大会的内容和价值，参会者就会从繁忙工作出抽出时间参加这个会议。如果把大会的日程安排在 2 天左右，那么大会是应该独立举办比较好，而不要附加在其他会议上举办，否则，时间跨度太大。

大会举办地放置在中立的地方，而不是放在厂商或者产业界那里举办，将会比较可行。目前已经有比较明显的举办地选择倾向，大多数与会者认为下一次会议应该放在 San Francisco Bay Area 举办，这样可以让大多数参会者节省路途上的时间。我们可以再次在 SLAC 开会，当然，Asilomar 也是一种可能的选择。

参会者的数量不要有明显的增加，因为，参会人数太多，就会让会议很难取得成果。还应该采用邀请的方式参会。

下一次大会的内容，应该侧重于经验分享，从而可以找到一些共性，并把这些共性发展成为群体范围内的广泛需求。如果数据库厂商参会，用户将会提出更多的相关问题。

6.2 工作组

本次大会的讨论定位在较高的层次。与会者认为，我们应该深入了解具体问题。一个经常被提及的例子是，学术界很愿意更多地了解科学界的需求。为了解决这些问题，比较好的方式是设立一个较小的工作组，成员之间可以经常见面交流。科学界/计算机科学界之间的见面，需要探讨的内容包括：

- 为科学数据库开发一个公共的需求集合，包括难度大的查询、用户认为比较理想的几种数据类型以及几种代数操作；
- 为科学界开发一套方法，从而能够让学术界界访问科学界的一些大型数据。

6.3 测试基准

定义良好的测试基准，是描述问题的一种很好的方式，可以吸引数据库厂商和学术界的注意力，推动这个领域的发展。现有的测试基准，比如 TPC-H、TPC-DS 是比较有用的，但是，它们并没有面向超大规模数据库。工作小组将会首先在公共需求方面达成一个共识，然后定义一个专门针对数据密集型查询的测试基准。

6.4 共享基础设施

如果不同群体可以避免一些重复工作，尤其是不要重复错误，那么这个领域的进展就会很快。这里，共享基础设施将会带来很大的收益。最初，我们将会为不同群体创建一个 wiki 站点，发布一些项目的经验教训，描述并探讨一些问题。这个站点规模会比较适中，但是，会面向所有感兴趣的群体，包括那些没有机会参加大会的人。

与会者也注意到，我们应该尽力利用最近发布的数据中心，这个数据中心本来是为学术界提供服务的，它采用 Google 硬件、IBM 管理软件和 Yahoo!主导的开源 map/reduce 软件。

7 致谢

大会组织者非常感谢来自以下赞助商的支持：LSST 公司和 Yahoo!。

8 词汇表

CERN - The European Organization for Nuclear Research
DBMS – Database Management Systems
ETL - Extract, Transform and Load (data preparation)
GFS – Google File System
HEP – High Energy Physics
IPAC - Infrared Processing and Analysis Center, part of the California Institute of Technology
JHU - The Johns Hopkins University
LHC – Large Hadron Collider
LLNL - Lawrence Livermore National Laboratory
LSST – Large Synoptic Survey Telescope
NCSA - National Center for Supercomputing Applications
OLTP - On-Line Transaction Processing
ORNL - Oak Ridge National Laboratory
PanSTARRS – Panoramic Survey Telescope & Rapid Response System
PNL - Pacific Northwest National Laboratory
RDBMS – Relational Database Management System
SDSC - San Diego Supercomputer Center
SDSS – Sloan Digital Sky Survey
SLAC – Stanford Linear Accelerator Center
VLDB – Very Large Databases
XLDB – Extremely Large Databases

附录-大会日程表和参会者名单（略）

(全文完)