

知识库系统及其研究现状

周傲英 (复旦大学计算机系)

摘

要

Knowledge-base technology has grown up to one of the mainstreams of nowadays database technology. The theoretical study and system implementation about knowledge-base technology have been receiving great attention all over the world. In this paper, we give a brief overview about several typical experimental knowledge-base systems, such as NAIL, LDL, NU-Prolog, KIWI, EDUCE, etc., and their characteristics are extracted. Then, the development trend of knowledge-base systems and the relevant technology are discussed. Furthermore, some ideas about our future research are described.

一、引言

数据库技术自出现以来一直以惊人的速度发展。它在商业应用中的成功奠定了其重要地位。七十年代后期,人们已不满足于对关系数据库的研究,很多学者试图从逻辑的观点出发,以数理逻辑作为理论基础来研究数据库。这一尝试导致了八十年代前半期演绎数据库的迅速发展^[1]。这一时期的研究成果为新一代数据库技术的研究和发展铺设了一条道路。

几乎是在数据库技术取得交口赞誉的同时,人们就已注意到了这些传统技术的局限性。随着计算机技术在各领域应用中的推广和深入,对更强功能的信息管理技术的要求越来越迫切,传统数据库技术的局限性也就显得越来越严重。在新的应用需求中,人们首先注意到的是非商业应用系统(如VLSI CAD, CASE及模式识别系统等)的需求。这些应用中信息的一个典型特点是结构复杂(层次、嵌套、递归);对管理(如更新、查询)有特殊要求,因此八十年代后期面向对象数据库(OODB)和知识库(KB)应运而生。有人预言,九十年代将是OODB的时代或是

KB的时代。这说明了人们对研究这些新技术寄予了很大希望。至于哪种技术将成为取代七十年代以来风靡一时的RDB技术还有待于实践来裁决,抑或新一代数据管理技术将融汇现在人们观念中的OODB、KB乃至其它技术。本文将讨论当今知识库系统(KBS)的研究现状及发展趋势,为研制KBS及相关技术提供一点参考。

关于知识库系统, D. H. Warren是这样来描述的, KBS是一个“…能有效处理中等大小的知识库(由三千个谓词,三万条规则,三百万个事实组成,总存储量为30兆字节)的逻辑程序设计(LP)系统”。^[2]很显然,他是从LP角度出发的。J. D. Ullman还强调了KBS类似于数据库系统的功能。他认为一个KBS可看作是一个具有以下两个特征的LP系统^[3]。

1. 有一个描述性语言,既可用作宿主语言又可作查询语言。

2. 支持数据库系统的主要功能,如大批量数据的高效存取、数据共享、并发控制及错误恢复等。

相比之下,Ullman的描述更具体,已被广

泛接受。可是，直到现在还没有一个具有这样特征的商品化KBS^[2]。

国际上许多著名的研究机构和大学实验室正在设计或实现这样的KBS。一些实验性系统已初具规模^[3]。继Ullman 1985年发表其著名论文“数据库逻辑查询语言的实现”^[4]以来，斯坦福大学在KB语言的语义表达能力和KB查询优化方面做了许多杰出的工作。同时，他们还开发了KBS实验系统NAIL₁，不仅从系统功能上探讨了KBS的实现，而且还能及时地对最新研究成果进行实用性验证。美国MCC集团的LDL系统以其雄心勃勃的计划和独特的设计思想一直吸引着人们的注意，影响着KBS的研究和开发。

在欧洲，由欧洲三大计算机生产商(BU-LL, ICL, 和西门子)在1984年发起创立的联合研究中心ECRC所属的知识库组开发了EDUCE、DEDGIN、KB2等一系列原型系统。它们分别解决KBS中知识的演绎推理、概念建模、物理存储和有效性控制等问题。KIWI系统是由意、荷、法、比、丹麦等国的研究机构和大学组成的联合研究体承担的欧共体ESPRIT计划下的项目。该系统除具有当今KBS的典型特点外，还能支持面向对象的知识表达语言OOPS。墨尔本大学的NU-Prolog系统被认为是从逻辑语言出发研制KBS的典型代表。NU-Prolog具有DB系统的许多特征。它从各个层次(如查询语言、索引模式、连接设施、查询计值技术以及并发事务处理)探讨了集成逻辑推理机制和DB系统所产生的问题。

自1986年以来，国内数据库界在系统消化吸收国际先进成果的基础上广泛开展了对知识库的研究。“七·五”期间，复旦大学计算机系研制开发了通用的知识库系统KBASE，并于1990年9月通过机电部电科院主持的鉴定。KBASE系统基本上包含了LDL和NAIL₁系统的主要功能，并在很多方面有所扩展和创新^[5]。对KBASE的功能扩展、性能改善以及相关技术的研究正在进行

之中。

二、典型KBS的系统概貌

八十年代中期以来，在KB语言和查询优化等技术的研究上取得了很大进展，KBS实验系统从各方面探索了理论成果实用化的途径。KBS的这种发展模式与七十年代RDB的发展模式极为相似。尽管现有的原型系统功能并不完善，而且各有侧重，但各原型系统所取得的成果表明，商品化的KBS的推出已指日可待。

1. NAIL₁系统^[3,6]

NAIL₁系统是斯坦福大学J. D. Ullman主持下开发的KBS系统。其系统结构如图1所示。我们可以通过系统完成一个查询过程来简单介绍该系统。

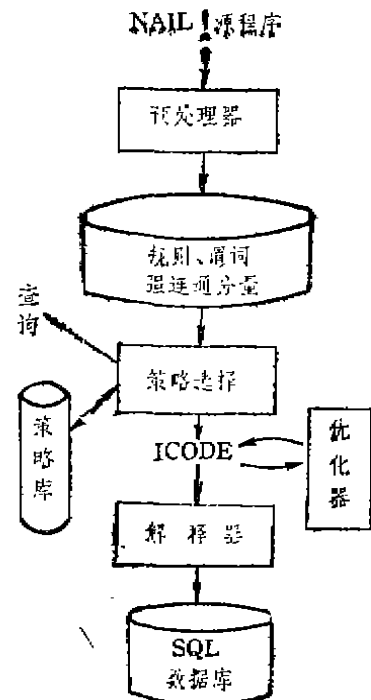


图1

系统的输入是描述应用系统的NAIL₁源程序和用户查询。NAIL₁语言是扩充的DATALOG语言。系统接受源程序后，预编译器把它编译成内部表，同时生成一张谓词图，把谓词划分成不同的强连通分量(SCC)。SCC是确定查询计值策略时所考虑的基本单

元。

策略选择子系统是NAIL₁的核心。它接受查询并为之选择最佳实现方案。这个实现方案是由一组为计值过程中涉及到的修饰目标(adorned goal)选定的计值策略粘接而成的,修饰目标表明哪些参量有给定值,哪些参量需求解。方案的选择过程是一个自顶向上试探回溯的规则/目标图(RGG)构造过程,并辅以一些启发式规则(如bound-is-easier)和避免重复选择的措施(策略库)。构造过程中,对每一个修饰目标系统要咨询在此之前积累起来的策略库为该目标的计值选择一条俘获规则(capture rule)。如果当前策略库不能提供这样的信息,系统则按序测试适合于该目标的一组俘获规则。俘获规则实际上对应于逻辑规则的计值方法,如“左线性递归法”、“魔集法(magic set)”、“计数法(counting)”等。对于不同类的SCC(如EDB类、非递归DATALOG类、递归DATALOG类…)可供选择的俘获规则也不同。为一修饰目标选定俘获规则后,还有可能要进行回溯,因为被选俘获规则又会产生出同一SCC中的谓词和下层(strata)谓词的修饰目标。如果其中有一目标的任何可选俘获规则都不适用,则向上回溯,改变上一轮的选择,不管该选择是咨询策略库所得还是任选的。这种回溯似乎是穷尽搜索,但策略库和启发式规则的使用改善了这种搜索。NAIL₁的策略选择算法是一个自顶向下而兼有自底向上优点的启发式算法。如果这个过程成功结束,即系统构造出完整的包含全都相关谓词的RGG,那么把图中修饰目标的俘获规则表示的计值策略粘接起来就形成了回答初始查询的计值方案。

在策略选择完成以后,系统就用中间语言ICODE来表示求解查询的算法。ICODE是扩充了的关系代数语言。优化器专门用来对ICODE进行优化。解释器是一个把ICODE程序翻译成SQL关系运算的接口子系统。

2. LDL系统^[3,5]

• 58 •

MCC的第一个实验性LDL系统主要由以下几个模块组成:用户接口、模式管理器、规则管理器、查询格式管理器、优化器、查询管理器。LDL系统的用户语言是LDL语言,是一种支持否定、集合对象、函数项的描述性语言。为了KBS的数据库系统功能的需要,它还支持一定的过程性。它代表了把逻辑语言和数据库语言有机统一起来的思想,在KBS的发展中前景大有希望。

规则管理器、查询模式管理器和优化器实现LDL系统的规则编译、查询优化和安全性技术。系统接受LDL程序(规则)后,规则管理器分析规则、产生其基于PCG的内都表示。查询格式管理器和优化器根据给定的查询格式(相当于NAIL₁中的修饰目标),对所涉及到的规则进行重写转换。优化器为查询模式构造一个处理图并进行简化,把SCC对应的部分压缩成一个结点,并为这样的结点标出按代价模型遴选出的不动点计算方法(实际上是SIP和递归处理方法)。非递归情形下的转换包括常量内迁;递归情形下的转换包括投影内推(存在参量识别)法、魔集法、左(右)线性法、计数法等。规则经改写转换后,可用半朴素(semi-naive)法求解其不动点,计值初始查询^[7]。经查询优化处理后,规则被翻译成FAD中间代码,继而转换成由关系代数运算表示的目标代码。查询管理器管理前述编译过程形成的目标模块并通过匹配实际查询和预编译的查询格式来选择执行合适的目标模块。

3. NU-Prolog系统^[5]

NU-Prolog是墨尔本大学继MU-Prolog以后开发的另一个系统。MU-Prolog在传统Prolog系统中增加了外存管理能力。用文件系统管理事实较好地解决了用Prolog设计数据密集型知识系统的效率问题。如果说MU-Prolog是纯粹从LP出发的,那么NU-Prolog则是在设计初期就确立了一体化逻辑/数据库系统,即Ullman所描述的KBS这个目标。

NU-Prolog的查询语言支持数据更新

(插入、删除)。为了保证多用户环境下的可用性,系统增加了并发控制设施:事务原语。NU-Prolog的文件系统利用了叠加码字(simc)技术来处理复杂项(含函子或变量)的存储和检索。这样,在外存上不仅可以管理事实,还可以管理单元子句甚至规则。在查询优化方面,NU-Prolog保留和(用超连接Superjoin技术)改进了传统的自上而下目标驱动求值法。系统中加入了自底向上计值方法。在递归和分层情形下,经NU-Prolog系统转换后的等价程序能够保证分层性和域独立性。对于非分层情形,NU-Prolog所用的新魔集算法可生成一个迭代方程组,其求解过程仍保持高效。

4. 其它KBS

ECRC知识库组的目标是开发能把AI系统的信息建模能力、演绎能力等和DB系统的大容量信息管理能力结合起来的系统。最早推出的EDUCE是基于工程观点为结合Prolog和RDBMS的能力而开发的,而早期版本基于MU-Prolog和INGRES,它以混合方式实现了Prolog和RDBMS的松耦合和紧集成。DEDGIN可以看作是在RDBMS上增强了查询/回答能力而形成的,它提供给用户的接口是一个纯说明性语言。其查询计值过程对用户是透明的,处理递归查询所用的是QSQ法和QoSAQ法。KB2提供了强有力的信息建模和语义完整性保证能力,其语义信息模型扩充了“IS-A层次”的ER模型,约束保证技术是由其完整性子系统SOUNDCHECK实现的。在KBS的物理组织方面,ECRC研制了一种称为BANG文件的“格文件”结构。SATCHMO实现了KBS中的知识同化吸收功能(如规则的一致性检测)。

ESPRIT的KIWI系统由三个层次组成:用户接口UI、知识处理器KH和数据库环境ADE。系统所支持的OOPS语言丰富了LP特征,是面向对象的知识表达语言。KH实现了OOPS的除事实管理外的所有特性。ADE则接受来自KH的处理事实的请求,它是基

于LP和RDB的结合而设计的,它支持的LP语言是扩充了的DATALOG(基谓词对应于对象而非关系)。ADE负责把对象系统映射成元组系统,它所用的编译优化技术是“查询/子查询法”和“辅助魔集法”。

三、现有KBS的特点

本节从系统功能、用户语言和优化技术等方面讨论现有KBS实验系统的特点。

1. 系统功能

KBS的两大特征是支持描述性语言和具有DB系统功能。从这个意思上来讲,现有的KBS实验系统都很不完善。要使KBS具有DB系统的功能,势必要在语言中增加非逻辑、过程性成分,破坏其描述性。然而,在KBS实用化进程中这种功能又是最起码和必不可少的。

LDL语言中包含有进行事务处理的最基本的两个谓词:事实插入和删除谓词。为了处理这两个谓词,系统增加了负责规则良构性及Church-Rosser特性检测的成份。NU-Prolog的事务原语具有良定义语义,且可保证一组插入和删除操作原子地完成。为了支持多用户,UCB的POSTGRES新增加了四种锁机制^[9]。由于KBS发展的需要,无论在理论研究还是在技术实现上,KBS的事务管理和并发控制都是亟待探索的领域。

NU-Prolog系统采用叠加码字技术存储规则和事实,这比元组索引技术更适合于KBS。ECRC的BANG文件结构也是为了满足KBS的特殊要求而设计的。近几年为部分匹配检索所研究的文件组织技术对此很有借鉴意义。其它诸如错误恢复和知识有效性控制等,现有实验系统都不曾或极少涉及。

2. 用户语言及其实现

由于Prolog的广泛被接受,对KB语言的研究很快聚焦在逻辑语言上。DATALOG和LDL语言相对于Prolog而言的最大改进之处是克服了其非逻辑机制带来的非描述性及其执行机制在数据密集场合的固有低效性。LDL语言支持逻辑意义上的否定子目标,支

持为增强语言的知识表达能力而增加的函数项和集合对象以及规则中的过程性信息。NAIL₁语言是一种扩充了否定和函数的DATALOG语言。KIWI的ADE支持的是另一种扩充的DATALOG。这些语言的使用克服了长期以来阻碍数据库发展的宿主语言和DML间的不匹配。此外,数理逻辑为研究这类语言的语义表达能力提供了坚实的理论基础;逻辑程序设计语言的研究为改善其实现效率提供了很多现成技术。可以预言,未来的KBS语言必定是在这类语言的基础上发展起来的一种语言。

在语言的实现方面,WAM模型在KBS中的不适应性是显而易见的。现用的执行模型大多是自底向上且基于关系代数的。处理查询时,LDL把查询和相关规则改写后编译成FAD代码,NAIL₁则在选定俘获规则后把规则翻译成ICODE代码。FAD和ICODE都是扩充了处理函数和控制流的关系代数语言。现有KBS实验系统之所以用RDBMS管理事实也是由这种执行模型决定的。执行模型中不应完全排斥类似于Prolog的求值方法。NAIL₁、LDL和NU-Prolog都是予以保留作为候选的计值方法。

3. 查询优化技术

自底向上计值的问题主要是执行计划的安全性和高效性。安全性是指计值过程的终止性和结果完备性。它基本上属于理论研究问题。对于任一计值策略都要研究其适用场合(包括递归类型和事实库特性)。NAIL₁就是依据这些研究来为不同类的SCC提供不同的可选俘获规则集。LDL系统在选择规则重写算法时也要考虑查询的格式和规则类型^[7]。

计值查询的效率是影响KBS实用的关键因素之一。自底向上处理递归查询时,狭义地从计值过程来看,其低效性主要来源于冗余计算和无效计算。半朴素方法是广泛使用的避免相关事实的重复计算的方法。克服无效计算的基本思想是利用查询中的常量来限

制无关事实的产生。最简单的情况是常量内迁。对于复杂情况,已有各种算法,[10]综述了这些算法,比较了它们的适用场合和运行效率。LDL系统用规则重写转换方式实现了计数法、魔集法和魔计数法。KIWI的ADE用的是辅助魔集法。ECRC的DEDGIN系统用的是QSQ法和QoSaq法。NAIL₁的策略选择子系统则是一个可随时扩充新方法的系统。新方法作为某些SCC类的可供选择的俘获规则。这种系统结构可以方便地使系统连续升级,使之处理查询时变得越来越“聪明”。

广义地来看,效率改善还有另外的途径。逻辑程序等价性的研究也为改善效率提供了广阔的天地。这方面的工作还处在理论研究阶段^[11]。

四、KBS及其相关技术的发展

本节,我们将讨论KBS的发展趋势和相关技术的发展。作为对上节有关内容的明确强调或补充。

1. KBS的发展趋势

从系统结构来看,J. D. Ullman描述的KBS的两点特征是最基本也是最具体的。它为未来KBS的研制开发工作指明了努力方向。

LDL和NU-Prolog尽管在事务管理上做了些尝试,但离KBS的要求还相差甚远。这方面的工作远比DB系统中相关工作的难度大,它涉及到逻辑语言的过程性语义扩展。KBS的并发控制机制的设计必须考虑到知识库应用系统的特殊性。

现有的实验性KBS系统都是基于RDBMS实现的。诸如事实库管理那样的工作由DBMS去做,而只着重研究查询优化技术、探索功能实现的可行性,采用这种方法不仅影响了系统性能,而且也阻碍了KBS其它功能的实现。因此,在商品化的系统中更紧密结合有自己的物理组织的数据库系统是必不可少的。开发原型系统,这种方法是可取的,而且不失为一个分散难点的好方法,

但这只是权宜之计。随着有关理论研究和
技术实现的全面铺开和突破, **KBS**最终将摆
脱现在这种依附于传统技术和系统的发展
模式, 而逐步走向有自己一套独立体系的发展
模式。

知识的同化吸收也是**KBS**的一个重要成
份。在**AI**领域, 这类工作早已开展。在**DB**
领域, 完整性约束的理论研究和具体实施
也一直和**DBMS**的实用紧密相关。**ECRC**在
这方面已有一些工作, 如**SATCHMO**和**SO-**
UNDCHECK系统。

KBS的开发环境也将在今后的研究中处
于一定的地位。开发环境的优劣直接影响**KB**
技术的应用和**KBS**的推广。支持面向对象语
言是研制好的开发环境的一条可取的途径,
如**KIWI**的**KH**层就支持面向对象语言**OOPS**。

2. 相关技术的发展

如上所述, 毫无疑问, 为**KBS**研究的物
理文件组织、并发控制机制, 事务处理等技
术将在今后几年里迅速发展。知识库语言和
查询优化研究的发展仍将是最近的。语言的
发展集中在两个方面: 一是扩展语言的过程
性成份, 如事务原语、并发控制原语等以支
持**KBS**的有关功能。二是扩展语言的其它逻
辑成份, 如否定、函数等以增强其知识表达
能力。这方面的工作早已开始, 今后的工作
主要是从实用角度对已有成果进行评价、实
现或开展新的研究。

查询优化技术的发展主要集中在对逻辑
程序等价性和含不完全信息的情况下的查
询处理的研究。对等价性的研究可以看作是
关系**DB**查询优化研究的继续。已有的工作包
括对递归的有界性、可线性化性及冗余性的
研究。一般说来, 对这些性质的确定都是**NP**
问题。为每一性质划出其对于整个情形集合
的**P**和**NP**边界将是今后研究追求的目标。至
今给出的判定条件大多是充分而非必要的^[11]。
对于**NP**情形, 启发式判定算法的设计对于查
询优化也将是有意义的工作。

魔集法, 计数法等为解决到底向上计值

的效率问题而提出的优化技术最初是针对
类似**DATALOG**程序那样的简单情形的。扩充
否定、函数等成分后, 如何推广这些技术以
解决新的计值问题已引起了人们的广泛兴
趣。这方面的研究正方兴未艾。这只是解决
问题的一种方法。对于新情况下的查询优化
问题, 今后肯定会推出新颖的技术。

从最近的发展来看, **KB**和**OODB**技术
的结合也逐渐形成一股潮流^[12]。因此, 在系
统实现到用户语言的各个层次上研究**KB**和
OODB技术的集成也是未来知识库发展的动
向。

五、结束语

本文介绍了国际上具有代表性的**KBS**实
验系统的系统概貌和它们的特点。这些系统
的主要目标各有侧重。它们探索了描述性
逻辑语言的实现和优化, 研究了实现**KBS**的
数据库系统功能和知识管理功能的途径。在
此基础上, 本文讨论了未来知识库系统及其
相关技术的发展趋势。其发展主要集中在对
系统结构的完善、系统性能的改善、知识同
化吸收技术的研究、与**OODB**技术的结合以
及开发环境的研制。下面以我们对今后在
KBS方面研究工作的初步设想作为本文的
结束语。

我们将紧密跟踪国际上有关课题的前
沿研究动态, 尽可能充分利用现有研究成
果和开发经验, 采取系统实现和理论研究
齐头并进、互相促进的方法在较高起点上
开展我们的工作。

在系统开发上, 我们拟采用当前国际
上惯用的分散难点、各个击破的方法。并
且为系统设计一个软总线式的可方便扩充
新技术的系统结构, 逐步实现**KBS**的目标。
第一阶段仍着眼于查询优化技术的实现及
其性能比较, 然后探索实现**KBS**的**DB**系
统功能的可能途径, 逐步过渡到具有独立
物理组织和各种独特机制的系统。最终目
标是建立实用的知识库系统。

在理论研究上, 我们将集中研究那些
至今国际上尚未取得满意结果, 而在**KBS**
的实

(转第3页)

——即依赖专家对规则进行修改的系统，在处理渐变方面不够高明。我们需要知道的是，当规则发生变化时，原先的神经网络以及它所存贮的经验数据是否仍然适用于变化后的情况。我们希望一个系统能反应这种变化。

WM 你的小组正在开发的过程控制应用指的是什么？

ER 这是一个与我们的一个股东合作的项目。我们以两种方式使用混合型系统。在制造和化工过程中，由于人们无法很好表达他们是如何用最少的材料和能源消耗来获得最大的生产量的，所以抽取规则就变得十分困难。不过，这些工厂拥有大量关于以往实际操作的数据，通过用这些数据进行训练之后，神经网络可以提供一种控制日常操作的理想方式。

WM 这没有考虑异常情况的发生。例如，当管道或阀门破裂时，你们怎样处理？

ER 这时，也许神经网络就不一定会作出正确的响应，因为用以训练它的常规操作数据库中并没有考虑这些异常情况，而且神经网络也不太善于处理它们预先不知道的情况。因而，我们就需要一个基于规则的系统。在遇到异常情形时，领域专家可以根据物理学、机械学和其它学科的原理想出解决的办法。比如，当某个阀门泄漏时，就应当关掉通向它的管道；当温度高于沸点时，就打开蒸气释放阀等等。

总而言之，神经网络多半处于运行状态，管理常规的操作，而基于规则的系统仅在发生异常情况时才进行干预。

WM 那么，你们将符号系统和神经网络结合起来的第二种方法是什么？

ER 即使在常规运行时，符号化模型也是很有用的，因为它能与领域专家交换信

息，能解释其推理机制。当然，在某些时候，领域专家也可以不采用它。所以我们可以使用神经网络来检测数据并从中抽取能够用于众多实际控制的符号化模型。

WM 看来，你们的研究工作是从实际应用的目的出发的。

ER 是的。有些应用已成功地使用了两种技术中的一种。从这样的应用开始是一个很好的策略。然后我们将考虑那些做得还不够好的部分，看看是否能把它们综合起来会更好一些。

WM 让我们来探讨一下体系结构的另一种技术即两院制结构。

ER 我们所指的两院制结构就是在整个系统中，大多数知识同时以神经网络和符号形式两种方式表示，每部分以各自独特的推理机制工作，但必要时可从一种形式中抽取知识并将其转换成另一种形式。所以，尽管知识是以两种形式表示的，但实质上是共享的。一个例子就是用神经网络去构造一个符号化模型，正如我们前面所描述的。当然我们希望反过来也同样如此。

比如，在保险业务领域中，我们希望当新的法则或规程制订后能相应地对神经网络进行更新。

WM 你对不远的将来有什么期望？

ER 我相信我们将会看到更多的这种混合型系统的具体应用实例，象我们已介绍过的制动和过程控制系统，也会看到用于构造这种系统的更好的工具的出现。同时，由于我们不断地探索如何最大限度地结合这两种技术，我们将有可能构造出更多的功能更强的系统。

〔梁戎刚 孙善晖译自《IEEE EXPERT》VOL.5 No.5, OCT.1990, 蔡庆生校〕

（接第61页）

用化过程中又急需解决的问题，如对逻辑语言进行扩展后的模型论语义及其合理的计算语义，分层或非分层递归的查询优化策略；

知识的同化吸收以及KB技术和OODB技术的有机结合等。（参考文献略）