

# 基于场景的面向方面软件体系结构度量方法<sup>\*</sup>

陈佳丽<sup>1</sup>, 张琳琳<sup>1</sup>, 张亚红<sup>1</sup>, 赵 楷<sup>1</sup>, 冯在文<sup>2</sup>, 倪友聪<sup>3</sup>

(1. 新疆大学 信息科学与工程学院, 乌鲁木齐 830046; 2. 武汉大学 软件工程国家重点实验室, 武汉 430072; 3. 福建师范大学 软件学院, 福州 350108)

**摘 要:** 软件体系结构的评估为提高软件质量、控制系统复杂性提供保证,但多数单纯基于场景或基于度量的评估技术普遍存在度量角度单一的问题。针对该问题,结合场景技术提出了一种支持面向方面软件体系结构的度量方法。通过一组度量指标对体系结构进行度量并量化其结构特征;引入场景更细粒度地刻画体系结构的质量属性,考察场景在体系结构中的映射程度。最后对某保险案例的体系结构设计进行对比与选择,验证了方法的可行性及实用价值。两种技术的结合为更全面地进行面向方面软件体系结构的评估提供支持与参考。

**关键词:** 软件体系结构; 面向方面体系结构; 场景; 度量

**中图分类号:** TP311      **文献标志码:** A      **文章编号:** 1001-3695(2013)02-0483-04

doi:10.3969/j.issn.1001-3695.2013.02.045

## Measurement method for aspect-oriented software architecture based on scenario

CHEN Jia-li<sup>1</sup>, ZHANG Lin-lin<sup>1</sup>, ZHANG Ya-hong<sup>1</sup>, ZHAO Kai<sup>1</sup>, FENG Zai-wen<sup>2</sup>, NI You-cong<sup>3</sup>

(1. College of Information Science & Engineering, Xinjiang University, Urumqi 830046, China; 2. State Key Laboratory of Software Engineering, Wuhan University, Wuhan 430072, China; 3. College of Software, Fujian Normal University, Fuzhou 350108, China)

**Abstract:** The evaluation of software architecture can improve software quality and control software complexity. However, there are several shortages in the most evaluation methods using either scenario or measurement technique alone. Based on scenario, this paper proposed a measurement method for aspect-oriented software architecture (named SMM4AOSA). SMM4AOSA measured the architecture by a suite of metrics and quantified its structural feature, provided a fine-grained description of the architecture by scenario, and then discussed how about a scenario was satisfied by a certain architectural design solution. A case study of a simple insurance system was demonstrated to show the feasibility and value of SMM4AOSA finally. This method can provide some helpful supports for evaluating aspect-oriented software architecture, as well as references.

**Key words:** software architecture; aspect-oriented architecture; scenario; measurement

## 0 引言

软件体系结构一般通过组件、连接件等元素对软件系统的结构和行为进行描述,是系统的高层抽象<sup>[1]</sup>。软件体系结构在系统开发早期阶段构建,其设计直接影响后续整个开发过程的各个阶段,体系结构的设计不当不仅会直接影响系统的质量属性,甚至还会引起巨大的经济风险。诸多研究人员认为根据问题描述和需求规约来评估系统的软件体系结构是该问题的主要解决方案。软件体系结构评估是一种确定体系结构属性及优缺点的技术或方法,保证该体系结构满足特定质量属性,并为开发者选择正确的体系结构提供参考<sup>[2]</sup>。

随着面向方面软件开发技术的深入发展,人们将研究重点从实现阶段转移到设计阶段,面向方面体系结构(aspect-oriented software architecture, AOSA)的研究已成为热点问题。国内外已经提出许多 AOSA 的设计方法,取得了一定的研究成果,然而针对面向方面体系结构评估的研究尚处于起步阶段,缺乏

系统全面的评估方法,因而寻求一种更系统的面向方面软件体系结构评估方法成为亟待解决的问题。

在软件体系结构评估领域,基于场景的评估技术与基于度量的评估技术备受关注,取得了较为显著的成果。虽然这两种技术的研究思路和研究内容不同,但它们之间并不矛盾,各有特色,如度量技术在基于场景的评估方法中可以作为评估手段来使用,场景技术在对软件体系结构的行为进行度量时则可以为度量提供信息,两者存在互补关系<sup>[2~4]</sup>。大多数软件质量属性极为复杂,很难用一个简单的尺度来衡量,而基于场景的度量具有一些不确定性<sup>[3]</sup>。因而本文认为场景可以弥补度量的不足,为体系结构的评估提供必要的上下文环境,两者的有机结合使得对质量属性的评估更有意义。

综上所述,本文针对基于场景和基于度量技术的不足,利用两种技术之间的互补关系,在 AOSA 的度量中引入场景,提出一种基于场景的面向方面体系结构度量方法。针对面向方面体系结构描述语言 AC2-ADL<sup>[5]</sup>的结构特征提出一组新的度

**收稿日期:** 2012-07-05; **修回日期:** 2012-08-19      **基金项目:** 国家自然科学基金资助项目(61100017);福建省自然科学基金资助项目(2012J01250, 2011J05146);新疆大学博士毕业生科研启动基金资助项目(BS090142)

**作者简介:** 陈佳丽(1986-),女,河南永城人,硕士研究生,CCF会员,主要研究方向为软件体系结构、面向方面软件开发(cjli101@126.com);张琳琳(1974-),女,副教授,博士,主要研究方向为面向方面技术、软件体系结构;张亚红(1987-),女,硕士研究生,主要研究方向为软件体系结构、Web服务。

量指标,不仅从组件的角度对体系结构的结构特征进行度量,还从场景的角度考察了场景在体系结构中的映射程度。

## 1 相关工作

鉴于不同的评估阶段、评估目的及评估质量属性,研究者们纷纷提出一些软件体系结构评估方法。文献[2,6]分别对基于场景、基于度量、基于经验及基于数学模型等评估方法进行了综述性的讨论。其中基于场景和基于度量的评估技术的研究较为突出,本文将对这两种技术的国内外研究现状进行讨论。

场景是风险承担者与系统简单交互的简短描述<sup>[7]</sup>。文献[8,9]分别对一些重要的基于场景的评估技术进行比较与分析,该技术旨在通过评估体系结构对场景的支持程度来发现潜在的风险,从而确定体系结构的质量属性。基于场景的软件体系结构分析方法(scenario-based software architecture analysis method, SAAM)<sup>[7]</sup>是运用较广泛的体系结构分析方法,适用于可修改性及可扩展性等质量属性。然而,SAAM只考虑了体系结构与间接场景的交互,缺乏对直接场景的评估。随着人们对体系结构评估技术的关注,众多体系结构评估方法不断涌现,ATAM<sup>[10]</sup>、ASAAM<sup>[11]</sup>以及ALMA<sup>[12]</sup>等都是在SAAM的基础上演化而来的。

国内一些研究人员也致力于基于场景的评估方法的研究。张莉等人<sup>[3]</sup>和刘霞等人<sup>[13]</sup>分别系统地综述了软件体系结构的分析方法,主要涉及了基于场景和基于度量和预测评估方法;孙昌爱等人<sup>[14]</sup>提出将SAAM运用到软件系统的体系结构质量属性的评估中;文献[15]对SAAM和ATAM进行了对比分析,提出将定量的度量与定性的场景相结合的思想,但缺少具体的实现方法。

自Chidamber等人<sup>[16]</sup>于1994年提出C&K面向对象软件度量方法以来,人们对其进行修改、扩展和精化,涌现出各种度量方法。度量对象也从最初主要集中于软件代码,逐渐扩展到面向方面编程软件<sup>[17]</sup>,甚至延伸到软件体系结构层<sup>[18,19]</sup>。文献[17]运用度量指标来推理面向方面程序中的故障;文献[18,19]分别展示了常规体系结构评估中的度量指标和度量工具。

对面向方面软件体系结构度量方面的研究相对比较少。主要有ASAAM<sup>[11]</sup>用于识别和定义面向方面体系结构中的方面,严格来说ASAAM是面向方面体系结构的重构而非评估。文献[20]提出将ASAAM方法与面向模块化的度量相结合,但其度量粒度还比较粗糙。文献[21]从组件和关注点的角度设计一组适用于度量AOSA的度量指标,主要包括关注点分离、耦合度、内聚度以及复杂性的度量。其中关注点分离指标只关注非功能属性,缺乏对功能属性的度量。文献[22]是基于类点(class point)和AC2-ADL的特征对面向方面软件体系结构的适应性度量。

鉴于上述软件体系结构评估技术的研究现状,本文在文献[21]的研究基础上,利用场景刻画某些质量属性,进一步从场景的角度对体系结构进行度量。由于体系结构的描述方法各不相同,因此有必要使度量指标的定义适用于体系结构描述方法的特定抽象。本文运用AC2-ADL描述语言对体系结构进行描述,制定一组基于AC2-ADL概念的度量指标,并通过实例来展示方法的应用。

## 2 基于场景的面向方面软件体系结构度量方法

为支持面向方面的体系结构评估,本文借鉴现有的度量技术,提出一种基于场景的面向方面软件体系结构度量方法(scenario-based measurement method for aspect-oriented software architecture, SMM4AOSA)。SMM4AOSA的主要思想是将场景引入到体系结构的度量中,通过场景描述体系结构中某些元素的交互程度,量化体系结构的评估结果,更细粒度地揭示方面体系结构对场景的支持程度,间接获得系统的质量属性。该方法的基本流程如图1所示。

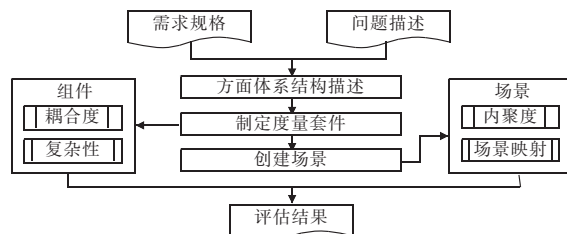


图1 SMM4AOSA方法基本流程

1) 软件体系结构描述 根据系统的需求和约束,采用特定的面向方面的体系结构描述语言对待评估的系统进行适当的抽象描述。除了传统体系结构中的组件、连接件以及两者间配置的描述外,方面体系结构中还应具备能够描述体系结构中横切关注点的方面组件、描述方面组件与其他元素之间关系的方面连接件等方面元素。该方法不仅适用于面向方面体系结构(AO)的度量,同时也适用于常规体系结构(non-AO)的度量。本文采用AC2-ADL分别对non-AO和AO体系结构进行描述。

2) 制定度量指标 依据特定的体系结构描述语言及评估人员对系统所关注的内容制定相应的度量指标。鉴于对场景的度量,通过修改和扩展部分度量指标,提出基于AC2-ADL概念的度量指标,具体定义如表1所示。

表1 基于AC2-ADL概念的度量指标

| 角度 | 属性   | 度量      | 定义                                                   |
|----|------|---------|------------------------------------------------------|
| 组件 | 耦合度  | Fan-in  | 由被评估组件提供服务的常规组件个数,以及影响被评估的组件的方面组件个数                  |
|    |      | Fan-out | 常规组件是为评估组件提供服务的组件个数<br>方面组件是被影响的组件个数以及为评估组件提供服务的组件个数 |
|    | 复杂性  | 接口个数    | 每一个组件的接口个数;常规组件(请求、提供接口);方面组件(请求、提供、横切接口)            |
|    |      | 操作个数    | 每一个组件接口中的操作个数                                        |
| 场景 | 内聚度  | 组件内聚    | 由被评估的组件解决的场景个数                                       |
|    |      | 组件个数    | 被评估场景调用的组件个数                                         |
|    | 场景映射 | 接口个数    | 被评估场景调用的接口个数                                         |
|    |      | 通知个数    | 被评估场景调用的所有方面组件中通知的个数                                 |
|    |      | 消息个数    | 被评估场景中传递的消息个数                                        |

由这组指标可以看出,从组件和场景两个角度对体系结构进行度量。其中组件的度量是对体系结构整体属性的测量,通过Fan-in、Fan-out、接口个数等度量指标衡量体系结构的某些属性,如耦合度、复杂度。另外由场景来体现系统的某些质量属性,并计算特定场景所调用的组件、接口、消息等个数,获得系统对场景的支持程度。

3) 场景的创建与描述 场景创建一般由软件风险承担者通过头脑风暴根据不同的评估目标选择出能够体现相应质量属性的关键场景。这些场景反映了他们对系统的需求,也体现了系统必须支持的各种活动。为了利用更丰富的信息来描述

场景在体系结构中的映射,采用序列图(sequence diagram,SD)来描述场景,一个序列图代表一个场景。

4) 体系结构度量 根据上述度量指标和场景对单个或多个体系结构进行评估。组件的度量结果量化地表示了整个体系结构的结构特征。场景的度量计算系统中单独的场景到体系结构元素(组件、接口、操作、消息等)的映射关系,换言之就是体系结构对每个场景的支持程度。

5) 总体评估 综合上述度量结果,评估人员需对体系结构进行总体权衡和评估,指出体系结构设计的不足、系统开发中可能存在的风险,甚至可以包括软件体系结构设计的改进建议等。根据不同的评估目标与需求,给出相应的整体评价。针对同一系统的多个体系结构方案,则可以通过比较度量结果来确定最终的体系结构。

### 3 案例研究

本章通过 Simple Insurers 保险公司提供的-一个保险业务系统<sup>[23]</sup>来分析该方法在实际中的应用。系统提供人寿、汽车和家庭保险三种类型,通过监听器对客户和保单实施跟踪,试图降低业务风险。系统业务人员能够为客户添加相应的保单,对参保客户、保单等进行一些基本操作,任何保单的变化都将向客户发出通知。

根据上述方法的步骤,首先由图2给出采用AC2-ADL描述语言对该保险系统进行描述的体系结构局部视图。该图呈现了两种不同体系结构解决方案,其中不包含阴影的方面组件部分与整幅图分别表述 non-AO 和 AO 体系结构。

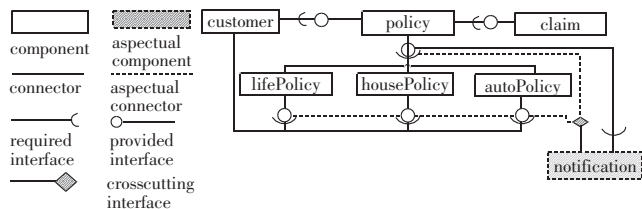


图2 保险系统体系结构局部视图

下面将选择性地对上述两个体系结构度量进行阐述。在 non-AO 方案中, customer(C1) 组件的体系结构 Fan-out 值为 4, 它分别向 policy(C2)、lifePolicy(C3)、housePolicy(C4) 和 autoPolicy(C5) 组件请求服务; Fan-in 值为 0, 因为它并不向任何组件提供服务。相比之下在 AO 方案中, 方面组件 notification(AC1) 横切(影响)了多个组件, 同时向 policy 组件请求服务。

由表2不难看出, non-AO 和 AO 方案的度量值并没有特别明显的差异, 其微小的区别取决于新增的方面组件 AC1, 以及受该方面组件影响的个别组件。系统往往追求的是高内聚、低耦合、低复杂度, 从这一角度来看, 新增的方面组件似乎不但没有降低其耦合度、复杂度, 有些值反而变高了。究竟哪个方案的体系结构比较合理呢, 下面将通过引入的场景度量来解答。

表2 保险系统体系结构度量部分结果

| 组件  | Fan-in |    | Fan-out |    | 接口个数   |    | 操作个数   |    |
|-----|--------|----|---------|----|--------|----|--------|----|
|     | non-AO | AO | non-AO  | AO | non-AO | AO | non-AO | AO |
| C1  | 0      | 0  | 4       | 4  | 4      | 4  | 22     | 22 |
| C2  | 2      | 3  | 1       | 2  | 3      | 3  | 25     | 25 |
| C3  | 1      | 1  | 1       | 2  | 2      | 2  | 9      | 9  |
| C4  | 1      | 1  | 1       | 2  | 2      | 2  | 7      | 7  |
| C5  | 1      | 1  | 1       | 2  | 2      | 2  | 9      | 9  |
| C6  | 1      | 1  | 0       | 0  | 1      | 2  | 15     | 15 |
| AC1 | 0      | 4  | 0       | 1  | 0      | 2  | 0      | 2  |

系统要求采用监听器对保单实时跟踪, 风险承担者(包括用户、开发人员、系统管理人员)根据该需求给出下列场景。S1 表示添加宠物保单; S2 表示保单到期提醒; S3 表示查看用户的保单; S4 表示保单变更时通知用户; S5 表示保单到期时间应小于保单创建时间。由于篇幅限制, 将只讨论保单变更的跟踪, 即只对场景 S4 进行评估。

S4 场景涉及多个保单的变更, 为了方便理解, 先分析单个保单的变更。假定某用户由于搬家只需变更 housePolicy, 该场景的 non-AO 和 AO 方案的顺序图分别如图3和4所示。通过分析可以看出, 不同方案的消息个数也不同。

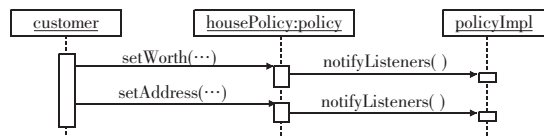


图3 housePolicy 场景变更顺序图(non-AO 方案)

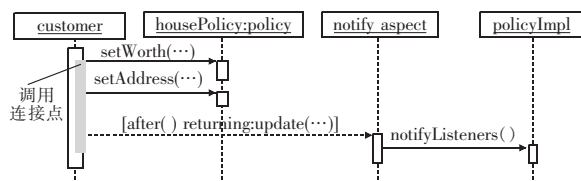


图4 housePolicy 场景变更顺序图(AO 方案)

在 non-AO 方案中, 修改 housePolicy 时要对其属性进行操作, 即分别调用 setWorth() 和 setAddress(), 该场景涉及到 C1、C2 和 C4 这三个组件。为了跟踪保单的改变并通知客户, 两个 set \* () 中均要实现对监听器的调用 notifyListeners() 以捕获保单的变化(图3)。而在 AO 解决方案中, 只需在方面中定义要捕获的连接点(保单变更的位置), 通过一个 after advice 即可实现对所有保单的监听, 包括对 policy、lifePolicy、autoPolicy 保单变更的监听(图4)。

在 S4 场景下, 用户可能参加多个保单, 一旦某用户的所有保单发生变化, 则需调用监听器对所有保单的属性实时跟踪。为保证对所有保单变化监控, customer 需分别向四个保单组件发出请求, 共涉及五个组件。而 AO 中除了这个组件外还新添了 AC1 方面组件。

如表3所示, non-AO 方案中, 组件 C2、C3、C4、C5 中对 notifyListeners 的调用个数分别是 8、3、2、3, 总计 16 个, 产生的消息个数为 32。AO 方案中, 新增的方面组件使得 policy 层次中不再调用 notifyListeners, 而被 notification aspect 中的 advice 单一调用所取代, 利用一个切入点捕获保单发生变更的所有连接点(所有以 set 开头的方法)。因而 set \* 方法个数加上额外的一个通知和消息调用, 消息总个数为 18。

表3 场景 S4 的映射度量

| 场景 | 组件个数   |    | 接口个数   |    | 消息个数   |    |
|----|--------|----|--------|----|--------|----|
|    | non-AO | AO | non-AO | AO | non-AO | AO |
| S4 | 5      | 6  | 10     | 6  | 18     | 32 |

从以上分析可以看出, 引入场景的度量更好地体现出方面组件的意义。尽管 AO 方案中组件之间的耦合性及复杂性比 non-AO 方案要稍高一些, 但通过特定场景的分析与度量表明 AO 比 non-AO 方案更合理。这是因为连接件中的输出消息越多, 系统中的交互程度就越强, 组件之间的依赖关系也会越复杂。在对单个场景 S4 的分析中可知, 消息的复杂性较大, 需要引入方面组件对保单组件作进一步分解, 将对保单的跟踪看做是一个横切关注点通过方面来封装, 可降低系统消息复杂性。

此外,AO 方案的设计还可以避免因开发人员的疏忽而遗漏了对某个保单或某个属性的跟踪。由上述分析不难得出,AO 方案的体系结构设计比 non-AO 更为合理。

#### 4 结束语

软件体系结构的评估对系统的质量起着至关重要的作用。本文在现有的体系结构度量研究成果的基础上,利用场景技术提出一种支持面向方面体系结构的度量方法。运用面向方面体系结构描述语言 AC2-ADL 对某保险案例的体系结构进行描述,并提出基于 AC2-ADL 概念的度量指标,通过这些指标从组件和场景两个角度对该案例进行度量。案例研究进一步证明了本方法的可行性,为该方法的实际应用提供了一定的参考。

在未来的工作中,本文将进一步完善度量指标,探索该方法在其他质量属性中的应用,如可适应性、可扩展性等,从而使本方法更加合理、更加科学。

#### 参考文献:

- [1] 梅宏,申峻嵘. 软件体系结构研究进展[J]. 软件学报,2006,17(6):1257-1275.
- [2] BANANI R, GRAHAM N. Methods for evaluating software architecture: a survey, Technical Report 2008-545 [R]. Kingston: Queen's University, 2008:1-82.
- [3] 张莉,高晖,王守信. 软件体系结构评估技术[J]. 软件学报,2008,19(6):1328-1339.
- [4] 宋光宇. 软件体系结构度量综述[J]. 计算机与数字工程,2008,36(1):41-44,69.
- [5] 文静,应时,张琳琳,等. 面向方面的体系结构描述语言 AC2-ADL [J]. 计算机科学,2009,36(8):126-132.
- [6] BREIVOLD H P, CRNKOVIC I. A systematic review of software architecture evolution research[J]. Information and Software Technology, 2012,54(1):16-40.
- [7] KAZMAN R, ABOWD G, BASS L, et al. Scenario-based analysis of software architecture[J]. IEEE Software, 1996,13(6):47-55.
- [8] HEIKO K. Sustainability evaluation of software architectures: a systematic review [C]//Proc of the 7th International ACM SIGSOFT Conference on the Quality of Software Architecture. New York: ACM Press, 2011:3-12.
- [9] DOBRICA L, NIEMELA E. A survey on software architecture analysis methods [J]. IEEE Trans on Software Engineering, 2002, 28(7):638-653.
- [10] RICK K, MARK K, MARIO B, et al. The architecture tradeoff analysis method [C]//Proc of the 4th International Conference on Engineering of Complex Computer System. Washington DC: IEEE Computer Society, 1998:68-78.
- [11] TEKINERDOGAN B, ASAAM. Aspectual software architecture analysis method [C]//Proc of the 4th Working IEEE/IFIP Conference on Software Architecture. Washington DC: IEEE Computer Society, 2004: 5-14.
- [12] BENGTTSSON P, LASSING N, BOSCH J, et al. Architecture-level modifiability analysis [J]. Journal of System and Software, 2004, 69(1-2):129-147.
- [13] 刘霞,李明树,王青,等. 软件体系结构分析与评价方法评述[J]. 计算机研究与发展, 2005,42(7):1247-1254.
- [14] 孙昌爱,刘超,金茂忠. 基于场景的软件体系结构分析[J]. 计算机工程与应用, 2000,36(9):75-79.
- [15] 沈群力,刘杰. 基于场景的两种软件体系结构评估方法[J]. 计算机应用研究, 2008,25(10):3015-3017.
- [16] CHIDAMBER S R, KEMERER C F. A metrics suite for object oriented design [J]. IEEE Trans on Software Engineering, 1994, 20(6):476-493.
- [17] BURROWS R, TAIANI F, GARCIA A, et al. Reasoning about faults in aspect-oriented programs: a metrics-based evaluation [C]//Proc of the 19th IEEE International Conference on Program Comprehension. Washington DC: IEEE Computer Society, 2011:131-140.
- [18] 宋光宇,高晖. 软件体系结构度量工具的研究与实现[J]. 计算机应用研究, 2008,25(9):2700-2702.
- [19] GARCIA-MAGARINO I, COSSENTINO M, SEIDITA V. A metrics suite for evaluating agent-oriented architectures [C]//Proc of the 25th ACM Symposium on Applied Computing. New York: ACM Press, 2010:912-919.
- [20] SANT' ANNA C, GAMEZ N, GARCIA A, et al. General architecture evaluation process/metrics, AOSD-Europe Deliverable D85 [R]. [S. l.]: AOSD-Europe, 2007.
- [21] SANT' ANNA C, LOBATO C, KULESZA U, et al. On the quantitative assessment of modular multi-agent system architectures [C]//Proc of International Conference on Multiagent Systems and Software Architecture. London: Springer-Verlag, 2006:111-135.
- [22] YE Peng, NI You-cong, HU Ming. Research on measurement technique for evaluating adaptability of aspect-oriented software architecture [J]. Advanced Materials Research, 2011, 268-270:1307-1312.
- [23] CLEMENT A, COLYER A, HARLEY G, et al. Using eclipse aspectJ: your first steps [EB/OL]. (2005-01-21). <http://www.informit.com/articles/article.aspx?p=357692>.

(上接第482页)存储的融合,在功能上能够同时满足 OLTP 和 OLAP 的技术需求;并且通过原型开发和实验证明了其可行性和有效性,对行列混合存储数据库作了有益的探索。然而早物化技术在一定程度上限制了列存储的读性能,笔者将进一步研究以迟物化方式在现有行式数据库中融入列存储的方法,以进一步提高行列混合数据库系统的性能。

#### 参考文献:

- [1] 李超,张明博,邢春晓,等. 列存储数据库关键技术综述[J]. 计算机科学, 2010,37(12):1-7.
- [2] STONEBRAKER M, ABADI D J, BATKIN A, et al. C-store: a column-oriented DBMS [C]//Proc of the 31st International Conference on Very Large Data Base. [S. l.]: VLDB Endowment, 2005:553-564.
- [3] ABADI D J, MADDEN S R, HACHEM N. Column-stores vs row-stores: how different are they really? [C]//Proc of ACM SIGMOD International Conference on Management of Data. 2008:967-980.
- [4] ABADI D J, BONCZ P A, HAIZOPOULOS S. Column-oriented databases systems [C]//Proc of the 35th International Conference on Very Large Data Base. [S. l.]: VLDB Endowment, 2009:1664-1665.
- [5] MacNICOL R, FRENCH B. Sybase IQ multiplex-designed for analytics [C]//Proc of the 30th International Conference on Very Large Data Base. [S. l.]: VLDB Endowment, 2004:1227-1230.
- [6] 于胜利,张延松,王珊,等. 基于行存储型的模拟列存储策略的研究[J]. 计算机研究与发展, 2010,47(5):878-885.
- [7] BONCZ P, ZUKOWSKI M, NES N. MonetDB/X100: hyperpipelining query execution [C]//Proc of the 2nd Biennial Conference on Innovative Data System Research. 2005:225-237.
- [8] DAI Xiao-lei, LI Chun-qing. The application of materialization strategies on OLAP in column oriented database systems [C]//Proc of the 3rd International Conference on Communication Software and Networks. 2011:305-308.
- [9] IVANOVA M, KERSTEN M L, NES N. Self-organizing strategies for a column store database [C]//Proc of the 11th International Conference on Extending Database Technology. 2008:157-168.
- [10] ABADI D J, MYERS D S, De WITT D J, et al. Materialization strategies in a column-oriented DBMS [C]//Proc of the 23rd IEEE International Conference on Data Engineering. 2007:466-475.
- [11] SILBERSCHATZ A, KORTH H F, SUDARSHAN S. 数据库系统与概念 [M]. 杨冬青,马秀莉,唐世渭,等译. 北京:机械工业出版社, 2008:307-312.