

学校编码: 10384

分类号_____密级_____

学号: X2013230833

UDC _____

厦 门 大 学

工 程 硕 士 学 位 论 文

基于 J2EE 的银行会计引擎系统的
设计与实现

Design and Implementation of Bank Accounting Engine
System Based on J2EE

陈体万

指导教师: 杨律青副教授

专业名称: 软件工程

论文提交日期: 2015年9月

论文答辩日期: 2015年11月

学位授予日期: 2015年12月

指导教师: _____

答辩委员会主席: _____

2015年9月

厦门大学学位论文原创性声明

本人呈交的学位论文是本人在导师指导下,独立完成的研究成果。本人在论文写作中参考其他个人或集体已经发表的研究成果,均在文中以适当方式明确标明,并符合法律规范和《厦门大学研究生学术活动规范(试行)》。

另外,该学位论文为()课题(组)的研究成果,获得()课题(组)经费或实验室的资助,在()实验室完成。(请在以上括号内填写课题或课题组负责人或实验室名称,未有此项声明内容的,可以不作特别声明。)

声明人(签名):

年 月 日

厦门大学学位论文著作权使用声明

本人同意厦门大学根据《中华人民共和国学位条例暂行实施办法》等规定保留和使用此学位论文，并向主管部门或其指定机构送交学位论文（包括纸质版和电子版），允许学位论文进入厦门大学图书馆及其数据库被查阅、借阅。本人同意厦门大学将学位论文加入全国博士、硕士学位论文共建单位数据库进行检索，将学位论文的标题和摘要汇编出版，采用影印、缩印或者其它方式合理复制学位论文。

本学位论文属于：

1. 经厦门大学保密委员会审查核定的保密学位论文，于
 年 月 日解密，解密后适用上述授权。
2. 不保密，适用上述授权。

（请在以上相应括号内打“√”或填上相应内容。保密学位论文应是已经厦门大学保密委员会审定过的学位论文，未经厦门大学保密委员会审定的学位论文均为公开学位论文。此声明栏不填写的，默认为公开学位论文，均适用上述授权。）

声明人（签名）：

年 月 日

摘要

服务和效率已成为当今各家银行的核心竞争力，只有更好的服务，才能提高客户的满意度；只有更高的效率，才能创造更大的价值。

某商业银行参照面向服务架构体系(简称 SOA)，建设新一代核心系统，以此进一步提高工作效率，提升服务质量。

搭建起覆盖七层应用架构的 12 个应用平台，新一代核心系统就基于这 12 个平台进行开发。12 个平台主要有：P1 是渠道服务平台，P2 是员工服务平台，P3 是 workflow 平台，P4 和 P5 是 ESB 企业服务总线平台，P6-P8 是应用服务平台，P9 是数据仓库平台，P10 是报表平台，P11 暂时未实施，P12 是在线查询服务平台。

会计引擎系统作为核心业务和总账会计系统的业务桥梁，实现了系统间的低耦合性，满足业务上交易与核算分离的要求，以快速响应产品创新。会计引擎实现了参数管理、明细处理功能、批量处理功能、报表功能。系统采用 Java 技术作为开发语言，Oracle 11g 作为数据库，J2EE 作为技术框架。

本文基于软件工程的思想与理论，以银行业务发展为目标，首先阐述了会计引擎系统开发背景和发展现状。在此基础上，根据面向对象分析设计思想对需求进行分析，构造出会计引擎系统的整体框架；然后详细描述了系统的设计和实现，并列举了部分实现界面和应用程序代码；最后对系统进行了测试和总结，并对下一步的工作进行了展望。

关键词：会计引擎；J2EE；Oracle

Abstract

Today service and efficiency has become the core competitiveness of the banks. The better service, the higher customer satisfaction. The higher efficiency, the greater value.

In order to improve the work efficiency and the service quality, the commercial bank has built a new generation of core system, with reference to the service oriented architecture (SOA).

The bank has set up a 12 application platform covering the seven tier application architecture, and the new system is based on these 12 platforms for development. P1 is the channel service platform, P2 is the staff service platform, P3 is the workflow platform, P4 and P5 is ESB, P6-P8 is the application service platform, P9 is the data warehouse platform, P10 is the report platform, P11 is not implemented, P12 is the online query service platform.

To meet the requirements of business transactions and accounting separation and quickly respond to the innovation of the products, the accounting engines system has realized the low coupling between systems, as the bridge of the core bank system and the general ledger accounting system. The accounting engine system realizes the configuration of the parameters, the function of detail service, the function of batch service, the function of the report. It uses Java technology as the development language, Oracle 11g as the database, J2EE as the technical framework.

Based on the idea and theory of software engineering and the bank's development goals, it's first to describe development background, present situation of accounting engine. On this basis, it is to analyze requirements according to the object-oriented ideas of analysis and design. The system is constructed to the entire framework. Then the design and implementation of the system is described in detail, and part of interfaces and application codes is listed. Finally, the system has been tested and summarized, and it is prospected for the future work.

Keywords: Accounting Engine; J2EE; Oracle

第一章 绪论	1
1.1 系统开发背景.....	1
1.2 系统发展现状.....	1
1.3 本文内容安排.....	2
第二章 系统相关技术介绍	3
2.1 J2EE 技术简介.....	3
2.2 CONTROL-M 简介.....	4
2.3 iBatis SQL MAPS 简介.....	5
2.4 ORACLE 数据库简介.....	6
2.5 本章小结.....	6
第三章 系统需求分析	7
3.1 系统目标.....	7
3.2 功能需求.....	8
3.2.1 参数业务功能.....	8
3.2.2 明细业务功能.....	19
3.2.3 批量业务功能.....	21
3.2.4 报表.....	25
3.3 非功能性需求.....	28
3.3.1 安全需求.....	28
3.3.2 可靠性需求.....	28
3.3.3 性能需求.....	28
3.4 本章小结.....	29
第四章 系统设计	30
4.1 系统概要设计.....	30
4.2 系统功能模块设计.....	32
4.2.1 参数处理模块.....	32

4.2.2 明细处理模块.....	33
4.2.3 批量处理模块.....	42
4.3 数据库设计.....	47
4.4 本章小结.....	54
第五章 系统实现.....	55
5.1 开发及运行环境.....	55
5.1.1 硬件环境.....	55
5.1.2 软件环境.....	55
5.2 参数处理.....	55
5.3 明细处理.....	58
5.4 批量处理.....	64
5.5 本章小结.....	68
第六章 系统测试.....	69
6.1 测试环境.....	69
6.2 测试方法.....	69
6.3 测试用例.....	69
6.4 本章小结.....	75
第七章 总结与展望.....	76
7.1 总结.....	76
7.2 展望.....	76
参考文献.....	77
致 谢.....	78

Contents

Chapter 1 Introduction.....	1
1.1 System Development Background.....	1
1.2 The Development Status of The System.....	1
1.3 Main Contents and Arrangement.....	2
Chapter 2 Introduction to Relevant Technologies.....	3
2.1 J2EE Introduction.....	3
2.2 Control-M Introduction.....	4
2.3 iBatis SQL Maps Introduction.....	5
2.4 Oracle Introduction.....	6
2.5 Summary.....	6
Chapter 3 System Requirement Analysis.....	7
3.1 The Target of The System.....	7
3.2 The Functional Requirements.....	8
3.2.1 Parameter Business Function.....	8
3.2.2 Detail Service Function.....	19
3.2.3 Batch Operation Function.....	21
3.2.4 Report Forms.....	25
3.3 The Non-Functional Requirements.....	28
3.3.1 Security Requirements.....	28
3.3.2 Reliability Requirements.....	28
3.3.3 Performance Requirements.....	28
3.4 Summary.....	29
Chapter 4 System Design.....	30
4.1 Overall Design.....	30
4.2 System Function Module Design.....	32
4.2.1 Parameter Processing Module.....	32

4.2.2 Detail Processing Module.....	33
4.2.3 Batch Processing Module.....	42
4.3 Database Design.....	47
4.4 Summary.....	54
Chapter 5 System Impementation.....	55
5.1 Development and Running Environment.....	55
5.1.1 Hardware Environment.....	55
5.1.2 Software Environment.....	55
5.2 Parameter Processing.....	55
5.3 Detail Processing.....	58
5.4 Batch Processing.....	64
5.5 Summary.....	68
Chapter 6 Test of System.....	69
6.1 Test Environment.....	69
6.2 Test Method.....	69
6.3 Test Use Case.....	69
6.4 Summary.....	75
Chapter 7 Conlusions and Prospects.....	76
7.1 Conlusions.....	76
7.2 Prospects.....	76
References.....	77
Acknowledgements.....	78

第一章 绪论

1.1 系统开发背景

当今服务和效率已成为各家银行的核心竞争力，只有更好的服务，才能提高客户的满意度；只有更高的效率，才能创造更大的价值。

某商业银行参照面向服务架构体系(简称 SOA)，建设新一代核心系统，以此进一步提高工作效率，提升服务质量。由于 SOA 是面向服务、组件化的一种体系，灵活性很高。某商业银行把全行业务包装成各个组件，然后运用业务模型搭建信息系统^[1]。

搭建起覆盖七层应用架构的 12 个应用平台，新一代核心系统就基于这 12 个平台进行开发。12 个平台主要有：P1 是渠道服务平台，P2 是员工服务平台，P3 是 workflow 平台，P4 和 P5 是 ESB 企业服务总线平台，P6-P8 是应用服务平台，P9 是数据仓库平台，P10 是报表平台，P11 暂时未实施，P12 是在线查询服务平台。

1.2 系统发展现状

通过对某商业银行会计业务流程的深入研究，整合了原来分散在各个系统内部的会计分录生成和核算检查处理模块，增加统一的参数管理组件，开发出一套独立开放的、易于扩展的会计引擎系统。让总账业务从核心系统中剥离出来形成单独的总账系统，实现了传统核心账务和总账会计业务的第一次真正分离。

会计引擎作为核心业务和总账系统的业务桥梁，实现了系统间的低耦合性，满足业务上交易与核算分离的要求，以快速响应产品创新，同时支持多会计准则，多地域多时区的不停断处理，方便业务人员集中维护会计准则。

会计引擎系统主要提供以下功能：

- 1、通过消息模式和文件模式以接收相关各个产品系统提供的交易流水和分录，产生相应的会计分录，并进行必要的分录汇总以满足总账，管理会计及其他管理系统后续处理要求。

- 2、按照统一的、独立的会计核算规则，通过批量方式对分录进行平衡检查和对账，支持自动生成必要的会计分录，以保证引擎处理的可靠性和一致性。

- 3、提供参数配置模块以支持业务人员对核算规则集中维护要求。

1.3 本文内容安排

本文重点探讨了会计引擎的系统设计和实现方案，简介系统相关技术，并进行需求分析、系统设计和实现以及系统测试，最后总结和展望未来工作。

文章全部分成七章，内容安排如下：

第一章：阐述系统开发背景和发展现状。

第二章：简介系统相关技术。

第三章：进行需求分析，主要有功能性和非功能性需求。

第四章：实现系统设计，包括概要设计、业务功能模块设计和数据库设计。

第五章：系统编码实现，介绍了功能实现界面和主要代码等。

第六章：通过各主要功能测试用例对测试过程和结果进行描述。

第七章：系统总结和对未来的工作进行展望。

第二章 系统相关技术介绍

2.1 J2EE 技术简介

1、Java 简介

Java 是一款面向对象语言，简洁有效，具有高度可移植性。Java 平台由 Java 应用编程接口（API）和 Java 虚拟机（JVM）构成。JVM 使经过编译的 Java 代码在任何系统上都能运行。Java 的优点是与平台无关，可以实现“一次编程，到处运行”^[2]。

2、J2EE 简介和架构

J2EE 是一种技术规范，为开发人员提供了新的工作平台，定义了标准应用发体系结构和部署环境。在该体系结构中，开发人员的注意力可以集中在封装商业逻辑和业务规则上，所有与基础结构相关问题和底层分配问题都由应用程序容器服务器来处理^[3]。如图 2-2 所示为 J2EE 的体系结构。

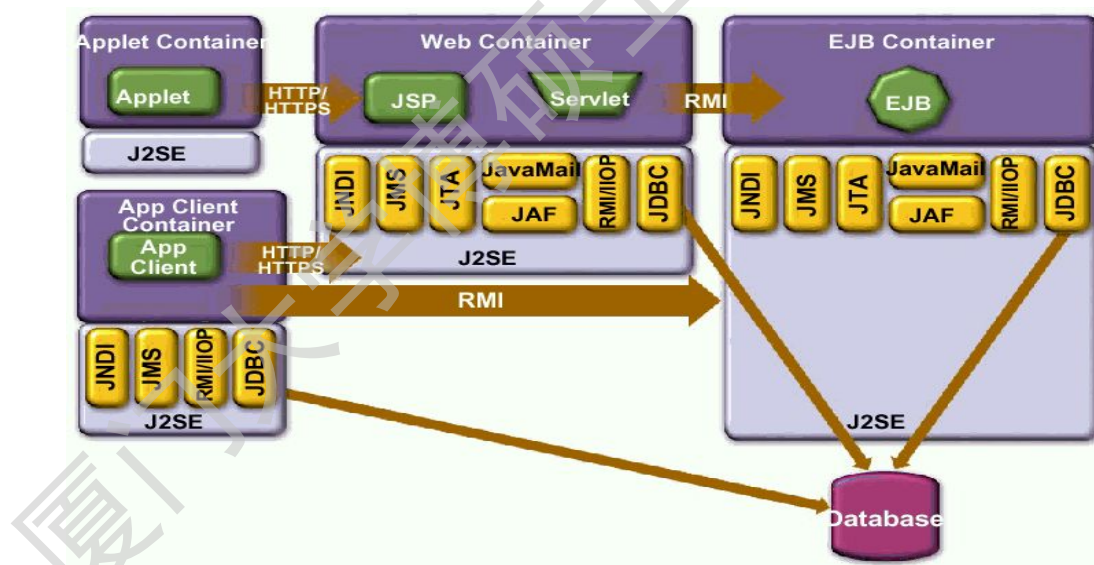


图 2-1 J2EE 架构

3、Eclipse 简介

Eclipse 是目前最流行的一款 Java 开发工具。它提供了一个框架和一组服务，通过插件组件构建开发环境。Eclipse 附带了一个标准的插件集，包括 Java 开发工具（JDK）^[4]。

2.2 Control-M 简介

Control-M 是 BMC 提供的批量作业调度的管理软件，应用 C/S 模式，把 Enterprise Manager 安装在服务器端，agent 安装在被控主机端，agent 在主机端提交由 CTM 定义好的作业流，并返回运行结果。EM 端可监控所有作业运行情况，可用多种方式来控制 Job 的运行条件和过程干预^[6]。

CTM 采用三层架构，包括 Enterprise Manager，Server 和 Agent 三个组件。EM 是图形化作业监控界面，用户可以看到作业流的执行状态，如果作业流执行过程有问题，可以通过 EM 干预和控制作业。通过 DESKTOP 定义作业和作业流，作业定义数据保存到 EM 后台数据库。EM 主要处理 EM GUI 和 EM 其他各组成部分之间的通信，执行数据库查询，为 EM GUI 提供数据。Server 是作业调度引擎，负责作业调度。Server 负责调度 Agent 作业运行，其自身也可执行作业，是 EM 调度和管理作业控制中心，负责调度作业、管理作业流、提供作业流状态的通知。Agent 负责作业的执行，安装在作业所处机器上，负责将作业提交给操作系统执行。Agent 是 EM 组成部分，服从 Server 管理，执行 Server 下发作业中指定的采集任务，并将结果返回给 Server 进行分析处理^[5]。

Server 负责调度作业，控制 Agent 进行采集工作，每个 Server 管理着若干个 Agent。Server 和 Agent 可以安装在不同主机上进行远程连接。图2-2 是 CTM 三层架构图。

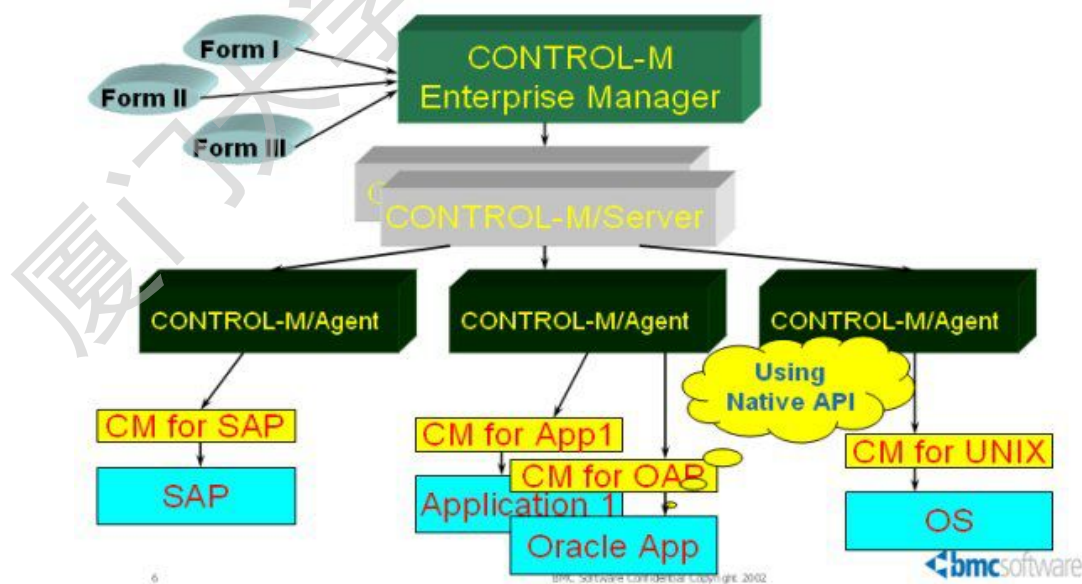


图2-2 CTM 三层架构

2.3 iBatis SQL Maps 简介

使用 SQL Maps，可以很大程度减少访问数据库代码。SQL Maps 提供简洁框架，使用 XML 描述文件将 Java Bean、Map 实现与基本的数据类型包装类（如 String）映射为 JDBC 的 PreparedStatement。SQL Maps 的生命周期如下：

将一个 Java Bean、Map 实现与基本的数据类型包装类对象作为参数，参数对象为 SQL 修改查询语句设定相关参数值。

(1) 执行 mapped statement。SQL Map 框架创建 PreparedStatement 实例，用参数对象对 PreparedStatement 实例设置参数，执行 PreparedStatement，从 ResultSet 生成结果对象。

(2) 执行 SQL 更新语句，返回更新数据行数。运行查询语句，返回一个 Java Bean、Map 实现与基本的数据类型包装类结果对象或对象的集合 [7]。

图 2-3 是 SQL Maps 执行流程图。

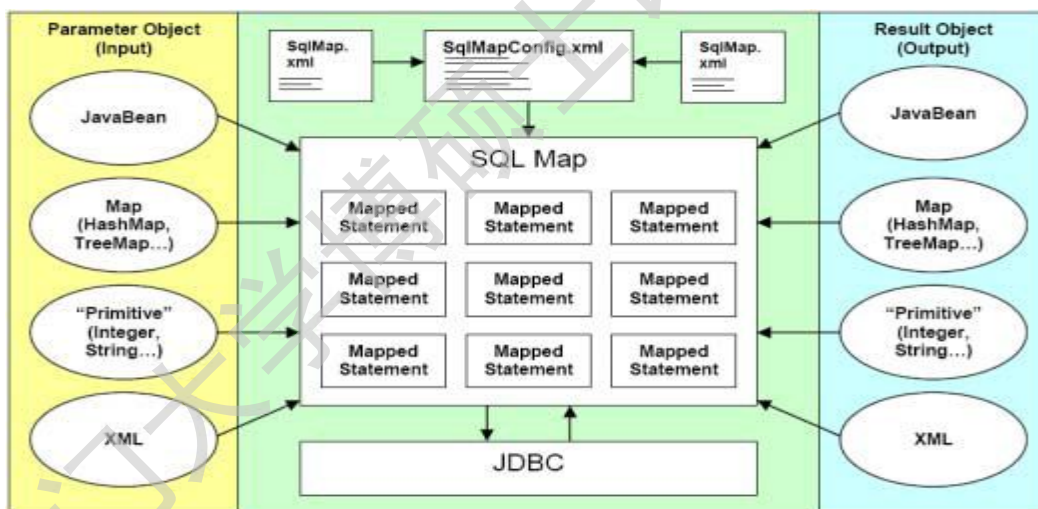


图 2-3 SQL Maps 执行流程图

2.4 Oracle 数据库简介

Oracle 是美国 Oracle 公司（甲骨文）提供的以分布式数据库为核心的一组软件产品，是目前使用最流行的数据库管理系统，作为通用数据库系统，它具有完整数据管理功能；作为关系数据库，它是完备关系产品；作为分布式数据库，它实现了分布式处理功能 [8]。

Oracle 数据库体系结构分为物理存储结构和逻辑存储结构。由于彼此相分离，所以在

管理数据的物理存储结构时并不影响存取逻辑存储结构。逻辑结构由至少一个表空间和数据库模式对象组成。模式是对象集合，模式对象则是直接的引用数据的逻辑结构。模式对象主要有：表、视图、序列、存储过程、同义词、索引、簇和数据库链等。逻辑存储结构包括表空间、段和范围，用于描述怎样使用数据库的物理空间。模式对象和关系形成了数据库关系设计^[9]。

Oracle 11g 数据库提供了高可用性、故障保护、回闪查询、回闪表、数据库和事务查询、数据卫士、Oracle 安全备份、服务器管理的备份和恢复等主要功能^[20]。

2.5 本章小结

本章描述了系统相关技术和调度管理工具，主要针对 J2EE、CTM、iBatis SQL Maps、Oracle 相关技术进行简要介绍。其中 J2EE 提供了项目技术框架；CTM 是统一监控和调度平台软件；Oracle 是后台数据库。

第三章 需求分析

需求分析是为了让技术人员和业务人员理解会计引擎系统具体的需求，主要功能性和非功能性需求等内容^[10]。

3.1 系统目标

会计引擎需要满足如下目标：

1、分录产生

(1) 全自动分录产生

会计引擎应该根据交易信息完全自动化产生会计分录，分录的产生不需要人工干预。

(2) 满足 COA 架构

会计引擎的分录必须满足 COA 架构中段值，包括平衡段和非平衡段。

(3) 总账输入的基准数据

会计引擎产生的分录进行汇总后作为总账的输入基准数据，除经费和固定资产外，总账只通过引擎获得财务数据。

2、参数化

(1) 业务可理解的分录产生规则

引擎的分录产生规则必须能被财务会计人员理解，满足会计准则和核算制度的要求，其规则的参数能直接由财务会计人员维护。

(2) 核算类型和核算服务目录定义

核算类型和核算服务必须能被财务会计人员理解，满足自然科目的设置，其目录的参数能直接由财务会计人员维护。

(3) 核算类型与产品类型、核算服务与原子交易映射

当产品系统增加产品类型或原子交易时，财务会计人员需维护核算类型和产品类型、核算服务和原子交易的映射关系。

3、信息回溯

(1) 平衡检查和对账

会计引擎在传输基准数据到总账前，完成借贷平衡检查。

会计引擎的机制和规则，能够支持总分核对、系统间往来的对账。

Degree papers are in the “[Xiamen University Electronic Theses and Dissertations Database](#)”.

Fulltexts are available in the following ways:

1. If your library is a CALIS member libraries, please log on <http://etd.calis.edu.cn/> and submit requests online, or consult the interlibrary loan department in your library.
2. For users of non-CALIS member libraries, please mail to etd@xmu.edu.cn for delivery details.