# 基于 J2EE 的日志管理中间件的设计和实现*

肖　文, 尹建伟, 陈　刚, 董金祥
(浙江大学 计算机科学与技术学院, 浙江 杭州 310027)

**摘　要**: 在分析现有大型系统对日志服务的需求的基础上, 针对现有的一些日志管理中间件中存在的使用或移植上的局限, 提出了一种基于 J2EE 的灵活、可靠、易用的日志管理中间件——LogService。通过 XML 配置文件实现日志记录的动态建模和配置, 通过使用连接到数据库的输出器将日志记录按照预定的格式解析后存入数据库的表格中, 方便日志审计管理的查看和维护操作。实践应用表明, LogService 提供了全面的机制来实现和满足大型应用系统从开发到实施全过程中的日志方面的需求, 为系统运行阶段的审计信息提供了高效、灵活的管理手段。

**关键词**: 日志服务; 数据库; Log4j; 审计; J2EE
**中图法分类号**: TP391　　**文献标识码**: A　　**文章编号**: 1001-3695(2004)10-0173-03

# Design and Implementation of Log Management Middleware Based on J2EE

XIAO Wen, YIN Jian-wei, CHEN Gang, DONG Jin-xiang
*(College of Computer Science & Technology, Zhejiang University, Hangzhou Zhejiang 310027, China)*

**Abstract:** After analyzing the requirements of large-scale system on generic log service modules, a flexible, reliable and convenient solution, LogService, is presented based on J2EE architecture in order to solve the drawbacks of existing log management middleware in using or migration. The XML format file is used to implement the dynamic modeling and dynamic configuration in LogService. And the JDBCAppender is used to store the log records, parsed by the record parser, into the database. So it is convenient for audit module to view and maintain the log records. It's shown that LogService provides an all-sided mechanism to satisfy the logging requirement from development stage to implement stage in a large-scale system. And it also provides an effective and flexible manipulating method for the audit management information in running stage.

**Key words:** LogService; Database; Log4j; Audit; J2EE

## 1　引言

在软件开发过程中, 日志管理是一个非常重要的部分。任何大型应用系统都有其特有的系统日志管理或跟踪的 API。在应用程序中添加的日志记录主要是基于三个目的: ①监视代码中变量的变化情况, 周期性地记录到文件中供其他应用进行统计分析[1]; ②跟踪代码运行时的轨迹, 作为日后审计的依据; ③担当集成开发环境中的调试器, 向文件或控制台打印代码的调试信息。在最近几年里, 不同的开发团体提出了各种各样的日志管理方案来帮助实现日志服务功能, 其中比较著名的有 Apache 的一个开放源代码项目——Log4j[2]; 由 Hewlett-Packard 和 Tivoli 开发的 ARM(Application Response Measurement) API 的实现以及一些面向通用系统的事件处理系统(Event Handling Systems), 如 CORBA Event Service, JINI Distributed Event Service 和 ECHO Event Service 等[3~6]。但以上这些方案都存在一定的不足, 主要有: ①Log4j 具有通用、灵活的特性, 但它不支持日志信息在不同输出源之间的自动转储; ②Log4j 输出的信息不能根据内容分段存储, 这样势必就带来了查询上的不方便; ③Log4j 对于分布式的应用支持不够完善; ④其他的一些实现虽然效率上具有很大的优势, 但是过于负责而且是相对独立的软件系统, 不利于其他软件系统与之集成, 不具有通用性。

本文在 Log4j 的基础上, 针对现有日志服务模块的不足, 提出了一种改进的日志管理中间件——LogService 来建立一套完善的日志系统: 通过 XML 配置文件实现日志记录的动态建模和配置; 使用连接到数据库的输出器将日志记录按照预定的格式解析后存入数据库的表格中, 便于审计管理模块的信息查询; 针对分布式环境下的应用, LogService 还提供了容错的机制[7], 以保证服务的可靠性。

## 2　LogService 体系结构设计

作为大型应用系统的底层通用模块, LogService 需要实现灵活的日志建模和日志记录等一些基本操作(插入、查询和删除等)。为了满足审计管理的需要, LogService 还需要支持日志记录的备份和恢复功能。因此, 在设计上考虑了支持多种不同的记录存储目标之间的相互转储以及自动转储功能。

如图 1 所示, LogService 主要包含以下几个组件: ①Log4j

收稿日期: 2003-10-08; 修返日期: 2003-11-11
基金项目: 国家"863"计划资助项目(2001AA 415220)

核心模块,实现基本的日志等级分类以及基本的一些接口;②对外的接口 API,封装并简化了日志的基本操作;③日志配置及日志内容的解析工具,实现日志的动态配置以及日志内容的解析和分类;④日志的输出格式定义模块允许用户自己定义输出的日志内容的格式;⑤日志的维护机制针对不同的输出器定义不同的维护策略。

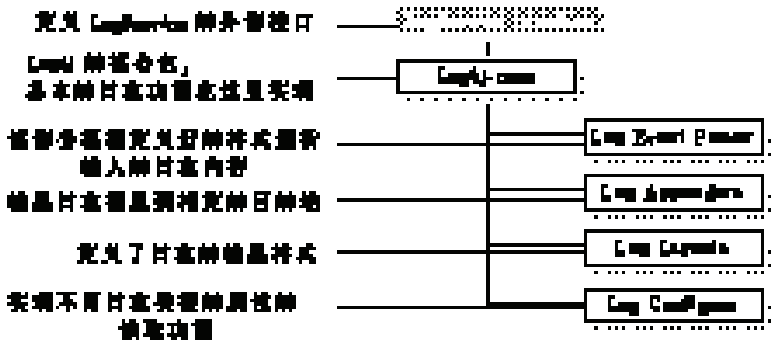

图 1 LogService 的体系结构框架

通过使用 LogService,开发者可以自己定义选择标准,对日志信息加以分类,依据信息类别和级别去记录信息,并能够在运行期间控制信息记录的方式以及日志存放地点;通过定义日志信息的级别,用户能够细致地控制日志的生成过程[2]。在 LogService 中还包含了一个简单实用的查询界面,可以方便用户从数据库或日志文件中查看、查询日志信息。

## 2.1 日志的自动建模

LogService 提出了日志的元模型的定义(图 2)。

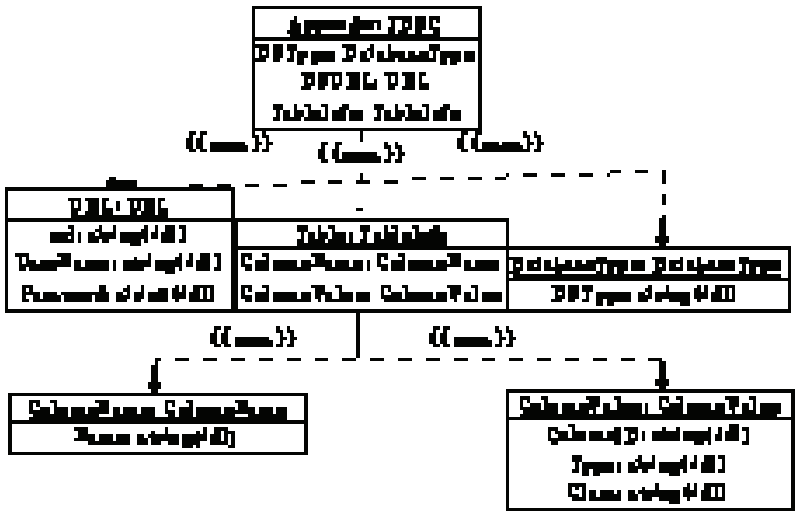

图 2 LogService 中的日志记录的元模型

在日志的元模型中定义了日志数据表中包含的元素:日志数据的数据库类型、日志数据库的连接信息、日志域定义的信息。其中日志域定义的信息包括针对某一列的数据、数据列的标志名称、数据类型的定义、数据内容的来源等。日志元模型的 XML 定义如下:

```
<! ELEMENT Appender #REQUIRED( #PCDATA) >
<! ELEMENT Database Type#REQUIRED( #PCDATA) >
<! ATTLIST Database Type value( mssq1 |oracle) #REQUIRED >
<! ELEMENT URL#REQUIRED( #PCDATA) >
<! ELEMENT TableInfo #REQUIRED( #PCDATA) >
<! ELEMENT ColumnName #REQUIRED( #PCDATA) >
<! ELEMENT ColumnValue #REQUIRED( #PCDATA) >
<! ELEMENT ColumnID #REQUIRED( #PCDATA) >
<! ELEMENT Type #REQUIRED( #PCDATA) >
<! ELEMENT Class #REQUIRED( #PCDATA) >
<! ATTLIST Class value( INC |LEVEL |LOGGER |THREAD |MSG) #REQUIRED >
```

## 2.2 日志配置及日志内容的解析工具设计

LogService 被设计成一套参数化的系统,如图 3 所示。

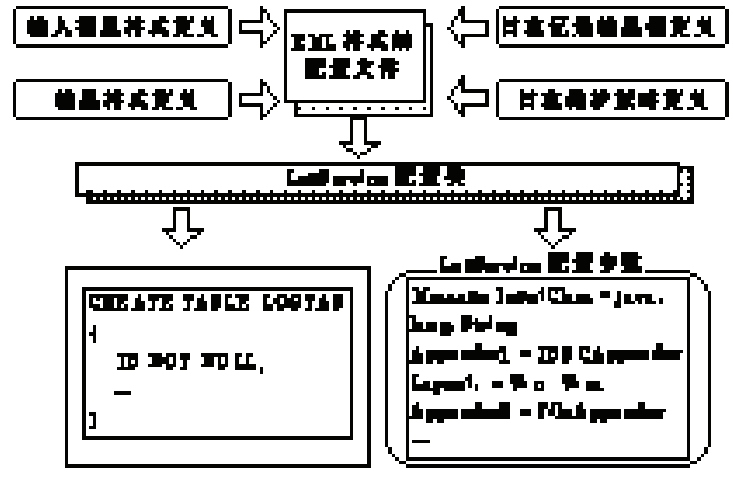

图 3 LogService 配置结构图

它的配置采用了 XML 文件,通过 XML 格式的配置文件所包含的丰富信息:日志的类别、不同类别日志的输入格式、输出格式、记录等级以及所采用的不同的输出器和输出目的等。

LogService 的配置类读取 XML 格式的配置文件对模块内使用的各个部分进行初始化,同时根据需要实现日志的建模。

日志的类别定义了包含不同属性的日志记录器(Logger)。不同的日志记录器可以采用不同的日志输入解析格式。通过读取配置,不同的日志记录器采用不同的方式解析输入的信息,利用对信息的检索分类,将经过处理的日志信息送往输出器(Appender)。日志记录器再根据配置的输出器,将日志信息输出到控制台以便于查看,或输出到文件、数据库等存储实体中以备日后的审计管理使用。

以数据库输出器(JDBCAppender)为例,当 LogService 初始化时,根据配置文件自动建立日志输入信息的解析格式,将输入的日志内容根据解析格式分解为不同含义的多个字段。在 LogService 将日志消息送往输出器时,输出器根据配置自动地将解析出的各个字段内容填入数据库表的相应列中。与以往的其他日志服务模块比较,在以 LogService 基础上开发的审计模块中,可以根据应用逻辑配置相应的日志格式,利用数据库的查询功能,建立符合上层应用逻辑相关的查询条件,查询符合条件的日志信息,极大地方便了审计管理的应用。LogService 同时提供自动建模的功能,能自动根据配置文件中的日志建模信息生成数据库的 DDL 脚本,或直接建立数据库表。

## 2.3 日志的维护机制

根据不同的输出器的特有属性,LogService 相应地建立了不同的维护机制来保证 LogService 正常有效的工作。主要包括:存储数据的备份、恢复、删除。

针对数据库输出器,LogService 实现了日志记录表的备份和恢复功能,如图 4 所示。

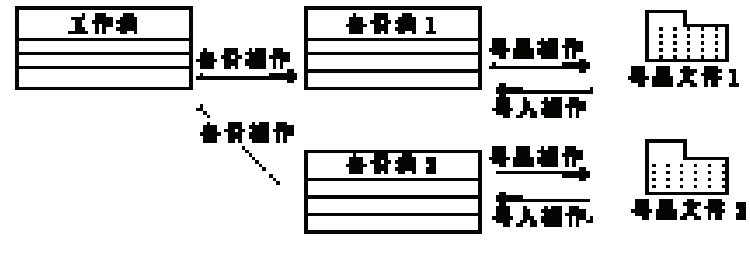

图 4 日志记录的备份恢复

用户可以将 LogService 的当前日志记录表中的内容转移到备份表中。LogService 还提供了从备份表导出到文件和从导出的文件恢复备份表的功能。备份表由系统自动命名创建,并且保留了备份时的日志类型信息,这样保证了将来恢复时的状

态一致性。

考虑到大型系统的日志记录为大数据量的操作, LogService 提供了自动转储功能。开发人员可以自己定义自动维护日志记录的规则, 规则中包括了自动转储的日志记录的数据源、转储数据的输出目的地; 转储的方式包括定期转储、记录到达一定数量之后等。

### 2.4　LogService 的其他特性

LogService 还针对分布式环境下的应用提供了容错的特性。当通过网络传送日志数据时, 由于存在网络的不稳定性, LogService 使用 "失败恢复 "的机制来实现容错。当网络连接在数据传送时发生, LogService 会将日志先记录在本地的一个缓存之内。在网络连接恢复后, LogService 首先检查缓存内是否有未发送的数据, 如果有则将其发送。

LogService 还提供了内存日志输出器( MemAppender) , 将日志内容建立在内存中的一块缓冲区中。它适合于需要临时日志内容的快速存取的场合, 比如文本编辑操作中的 Redo 和 Undo 功能的实现上。

基于自动建模功能的实现, LogService 还提供 "热配置 " ( Hot Configure) 的功能。它提供了一套触发器( Trigger) 来监听配置文件的修改, 实现了日志的动态配置: 可以在运行时通过修改配置文件来更改日志输出器的配置, 甚至可以在运行的时候进行日志的动态建模。

### 2.5　关键算法的实现

**算法 1　插入、查询和删除日志记录**

**输入: V - 待插入的日志记录**

Condiction—查询或删除的过滤条件

OpType—操作类型

```
{
PrepareConnection( ) ;  //准备数据库连接
switch( OpType) {
{
case INSERT:
MoveToInsertRow( RS) ;  //将记录集移动到新增的行
FormatMsg( V) ;  //格式化日志记录
Insert( RS, V) ;  //插入格式化处理后的日志记录
break;
case QUERY:
RS = QueryMsg( ) ;
break;
case DELETE:
ExecuteDelete( Condiction) ;
break;
}
}
```

**算法 2　转储日志数据记录**

输入: Table—待转储的日志记录

输出: BackupTable—转储生成的日志记录

```
{
QueryBackupTables( ) ;  //查询已存在的转储表
CreateBackupTable( ) ;  //创建转储表
TransmitLogRecord( ) ;  //备份日志数据
BackupLogTypeInfo( ) ;  //备份日志类型记录
}
```

## 3　应用实例

结合国家 "863 "课题 "基于 ASP 的网络化制造应用集成服务技术与系统 ", 我们实现了一个基于 J2EE 平台的审计管理系统, 审计日志记录数据建模结果如图 5 所示。该系统已应用在 "浙江块状特色经济区域网络制造系统研究与示范应用 " 系统中。

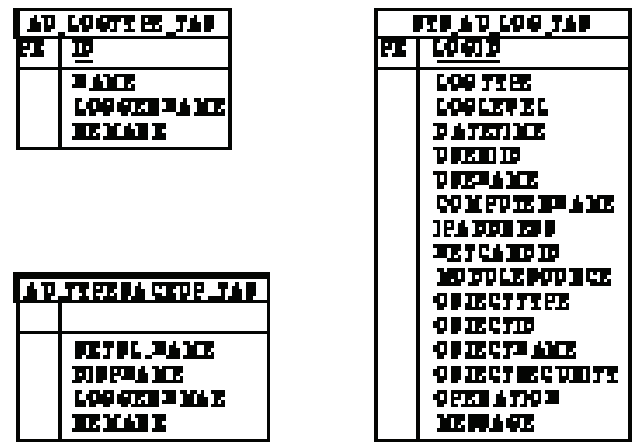

图 5　审计管理模块日志建模结果

## 4　结束语

在大型应用系统的开发和运行中, 系统的运行状况应该得到有效的监控和调整[8] 。为了获得高效、可靠和安全的运行过程, 通过系统的日志来进行记录和分析是一条现实可行的有效途径。实践证明, LogService 提供了全面的机制来实现并满足大型应用系统从开发到实施的全过程中的日志方面的需求。LogService 为系统开发调试过程中的调试信息的处理提供了有效的手段, 为系统的运行阶段的审计管理信息提供了高效、灵活的管理机制。

**参考文献:**


[ 1]　Ruffin. KiTLoG: A Generic Logging Tool[ C] . Proceedings of the ERCIM Workshops, INECS, Lisbon, Portugal, 1991. 139-146.

[ 2]　Log4j[ EB/OL] : http: //jakarta. apache. org/log4j/docs/index. html.

[ 3]　Open Group. Enterprise Management Forum[ EB/OL] . http: //www. opengroup. org/management/arm. htm, 2002.

[ 4]　CORBA. Systems Management: Event Management Service[ EB/OL] . X/Open Document Number: 437, http: //www. opengroup. org/online-pubs/008356299/.

[ 5]　Jini Distributed Event Specification[ EB/OL] . http: //www. sun. com/jini/specs/.

[ 6]　Eisenhauer G, F Bustamante, K Schwan. Event Services in High Performance Systems[ J] . Cluster Computing: The Journal of Networks, Software Tools, and Applications, 2001, 4( 3) : 243-252.

[ 7]　Tierney B, W Johnston, B Crowley, et al. The NetLogger Methodology for High Performance Distributed Systems Performance Analysis[ C] . Proceeding of IEEE High Performance Distributed Computing, 1998.

[ 8]　Tierney B, D Gunter, J Becla, et al. Using NetLogger for Distributed Systems Performance Analysis of the BaBar Data Analysis System [ C] . Proceedings of Computers in High Energy Physics 2000 ( CHEP 2000) , 2000.


**作者简介:**


肖文( 1978- ) , 男, 硕士研究生, 主要研究方向为 PDM、CIMS、数据库应用等; 尹建伟( 1974- ) , 男, 博士, 主要研究方向为 PDM、项目管理、工作流管理、数据库应用等; 陈刚( 1973- ) , 男, 副教授, 博士, 主要研究方向为 PDM、创新设计、信息安全等; 董金祥( 1945- ) , 男, 博士生导师, 主要研究方向为 CIMS、工程数据库、CAD 等。