

文章编号: 1005-8451 (2014) 03-0013-04

列车确报自动转发程序的开发

潘云松

(昆明铁路局 信息技术所, 昆明 650011)

摘要: 列车确报是指导铁路运输生产的重要基础信息, 确报转报是整个确报体系中是核心。为了提高确报转报质量, 昆明铁路局开发了确报自动转报程序。它利用调度计划和行车调度信息, 使用最短路径 (Dijkstra) 算法推算转报车站, 实现自动转报。文中重点论述程序设计原理和采用的技术。

关键词: 列车确报; 自动转发; 消息队列; Oracle OCCI

中图分类号: U285 : TP39 **文献标识码:** A

Department of train consist information auto pass-through program

PAN Yunsong

(Institute of Information Technology, Kunming Railway Administration, Kunming 650011, China)

Abstract: Train consist information was the important and basic information for railway transportation. Pass-through of train consist information was the core of Train Consist Information System. To improve the quality of pass-through, the Kunming Railway Administration developed train consist information auto pass-through program. The program used dispatching plan and train control information, utilized the improved Dijkstra Algorithm to calculate the pass-through station, implement auto pass-through. The paper described the design principle of the program and the used technologies.

Key words: train consist information; auto pass-through; Message Queue; Oracle OCCI

列车确报也称“运统一”, 是列车出发时向下一站和运输组织部门报告本列车及其车辆详细组成情况的信息报告, 是编组作业、卸车预报、车流调整等工作必不可少的信息, 也是铁路车辆追踪、行车调度、机车调度以及调车指挥作业的重要基础数据。

确报信息一般由编组站、区段站的确报系统产生。确报信息系统由铁路总公司、铁路局和车站3部分组成。车站部分主要实现确报发报和收报功能, 在铁路总公司和铁路局端主要实现确报转报功能。本文主要介绍铁路局端确报转报程序的开发。确报转报程序工作原理是根据确报报文信息, 将确报报文转发到需要的车站, 确报转报质量对车站车号人员的工作量和调车计划的编制至关重要, 需要通过考核提升确报质量。

1 确报转报现状和存在问题

铁路局确报信息系统主要由铁路总公司、铁

路局和车站3部分组成。车站是确报产生的源点和接收确报点, 铁路总公司和铁路局负责确报的转发。转发规则是: (1) 铁路局管内确报, 即发报站和收报站都是同一个铁路局, 确报由所在铁路局进行转发, 即发报站将确报报文传到铁路局, 由铁路局转报程序根据转报字典中的定义规则, 将报文转发到解体车站。(2) 跨铁路局报文的转发, 即发报站和到站(解体站)不在同一个铁路局, 转发规则是铁路局收到车站发报文, 判断到站是外铁路局车站的报文就上传铁路总公司, 由铁路总公司转报程序将报文转至到站所在铁路局, 再由铁路局转发到车站。

在实际工作中, 确报报文不仅需要转发到列车的终到站, 而且还需转发到途中经过的有甩挂作业的车站。目前, 铁路总公司确报转报系统已应用多年, 但还存在一些不足, 主要表现在:

(1) 转发报字典定义复杂, 确报漏发情况突出。在确报定义时要指明确报发站、到站、经由, 如果确报文中发站、到站、经由在转报字典中找不到就不会进行转报, 需要重新维护转报字典, 即便经常维护也难免疏漏。转报字典的记录数往

收稿日期: 2014-01-24

作者简介: 潘云松, 高级工程师。

往会达到上百条；在定义时还要考虑定义记录间相互影响，需要考虑程序发现第一个满足记录进行转报，本次转报结束，后面的定义都不会执行。

(2) 转报字典中定义的最大转报车站数不能超过5个。有些列车编组复杂，需要转报车站超过5个，转报时就会出现漏报，转报程序不判断是否需要转报（转报车站是否有作业），只要转报字典定义，就进行转报，造成车站接到许多没用的确报，给车号员挑选确报造成困难。

(3) 使用原有确报转报系统，许多车站收不到确报，经常需要手工转报，工作量大，不能满足24 h转报需要。

2 确报转报改进方案

为解决上述问题，昆明铁路局开发了确报自动转报程序，结合确报考核功能，提高确报发报质量。

当前铁路总公司所使用的确报传输软件是消息队列（MQ）软件，它基本可在所有系统平台安装，最大好处是不会丢失报文。确报自动转报程序中的重要模块如下。

2.1 确报分发程序

其作用是把发来确报MQ信息分发到两个应用队列中。1个队列中消息被铁路总公司程序接收入库，另1个被新确报转发程序处理，完成确报转发到车站的工作。

2.2 确报转发程序

其处理过程是：

(1) 解析XML格式报文，分析出报头中的发车时间、车次、发站、到站、经由信息以及报文中车辆到站信息。

(2) 根据确报的车次，从当前运行图中查找运行线，判断列车所经过的列车径路（列车所经过车站）。

(3) 如果不能找到到达终点站的整个路径，则再通过车次的计划线和基本图（基本上每天都要开行的车次），查找列车径路。

(4) 如果仍不能都找到，则根据发站、到站、经由，使用最短路径（Dijkstra）算法，找出发站到经由站列车径路和经由站到终到站的径路，求出它们合集，找出列车所经过车站。

(5) 从列车编组所有货车终到站集合中找出列车经过的车站，确定有进行作业车站，需要进行转报。

(6) 判断需要转报车站是否在确报站字典中（可能需要转报，但车站没有确报系统），确定需转报车站。

(7) 按照确定转报车站转发确报。

3 确报转发程序流程

昆明铁路局确报自动转报的程序流程如图1所示。

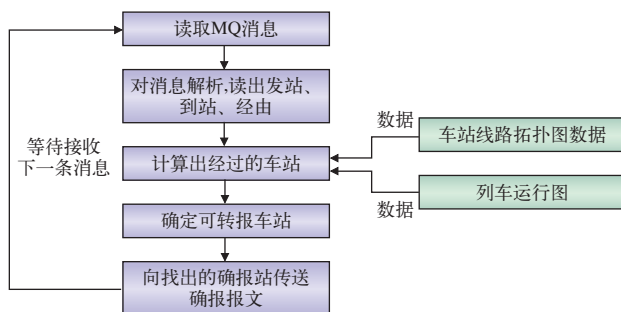


图1 确报自动转报程序流程图

(1) 车站确报点利用MQ通信软件，上传确报数据到铁路局，铁路局分发程序同时将MQ消息分发到2个队列中。

(2) 由铁路局入库程序将其中1个队列中的消息取出，并存入Oracle数据库。

(3) 新确报转报程序，对收到的报文进行合法性判断，对有问题的报文进行记录。

(4) 根据确报的发站、到站和经由，运用确报转报方案，找出转发确报的车站。

(5) 根据报文中车辆到站信息，推断出需要转报车站。通过MQ通信系统，将报文转发到车站。

4 系统实现

4.1 数据结构定义

关键数据结构采用C++语言描述，Oracle数据库存储，每条记录的存储以对象类型保存，程序对数据以对象为单元进行读取。

(1) 车站字典

记录结构是站名汉字、站名略码（是关键字）。

```
struct station
```

```
{
    std::string zmhzh; // 站名汉字
    std::string zmlm; // 站名略码
};
```

在数据库中的定义:

类型定义 create or replace type station

```
( zmhzh varchar2; -- 站名汉字
  zmlm varchar2; -- 站名略码
);
```

对应表定义 create or replace station_tab of station;

(2) 线路字典

记录结构是站名汉字、站名略码 (是关键字), 主要说明车起始站和终到站之间是否连同, 连通是有方向的, 终到站到起始站不一定是连通的, 该字典主要用于判定最短径路。

```
struct trainline
{
    std::string zm1;    // 起始站
    std::string zm2;    // 终到站
    float distance; // 站间距离
};
```

(3) 走行径路

两站之间走行径路, 经程序处理输出车辆结果, 存放列车径路。

```
struct NodeInf
{
    std::vector< std::string > path; // 经过车站
    float dist; // 走行距离
    NodeInf operator=( NodeInf a )
    {
        path=a.path;
        dist=a.dist;
        return *this;
    }
};
```

说明: 根据以上数据结构, 使用最短路径 (Dijkstra) 算法, 查找出列车经过车站。输入起始站和终到站, 输出结果以 NodeInf 的类型返回。

4.2 后台处理程序

考虑到程序的处理速度、稳定性和铁路总公司的现有传输体系情况, 选择 AIX 操作系统。AIX 是小型机上的一种操作系统, 比常见的 Windows 操作系统稳定, 代码执行效率高。

程序编写采用 MQ 中间件、Oracle 提供的 OCCI 和 XML 接口, 使用 C++ 语言编写 UNIX 后台程序, 实现对确报报文接收、解析和转发等功能。

以往 UNIX 后台程序是用 Oracle pro*c 编写: 需要先编写 *.pc 文件, 后生成 C 程序代码, 再进行程序编译, 程序代码编写和调试程序非常不方便。本文采用 Oracle 提供的 C/C++ 语言开发工具 OCCI(Oracle C++ Call Interface), 称为 Oracle C++ 语言调用接口。通过它, C++ 语言可以直接访问 Oracle 数据库的数据 (包括对象数据), 调用 Oracle 数据库的过程和函数。它的调用方式类似于 Java 通过 JDBC 访问数据库, 而运行效率远高于 JDBC, 因为它直接生成机器代码, 而 Java 编译生成的是中间代码, 执行时需要进行解释。能以对象数据为单元进行直接存取; 对 MQ 系统的调用采用 MQ for C++。经过 4 年使用证明, 该程序能很好地满足需要, 系统能在 2 ms ~ 10 ms 时间内完成一个确报转报工作。

(1) 后台程序在 AIX 系统后台运行情况

UID	PID	TTY	TIME	CMD
240	581768	pts/0	0:00	qb_resend
240	585868	-	0:00	rcv_data
240	700628	pts/0	0:00	ksh

(2) 后台转报程序处理情况

```
head = 22松林,宣威,41012,宣威,
TO.Q_HWM
TO.Q_XWM2
TO.Q_LJM
TO.Q_YEM
TO.Q_GYM

head = 22广通,大理东,45547,楚雄西,
head = 22田东,威舍,10875,威舍,
TO.Q_WSM2

head = 22昆明东,读书铺,46120,读书铺,
TO.Q_KXM2
TO.Q_DPM2

head = 22广通,甸尾,45264,甸尾,
TO.Q_DRM

head = 22广通,甸尾,45284,甸尾,
TO.Q_DRM

TO.Q_CXM
TO.Q_EXM
TO.Q_DOM
head = 22宣威,六盘水南,22626,凤凰山,

head = 22松林,宣威,41012,宣威,
```

TO.Q_HWM
TO.Q_XWM2
TO.Q_LJM
TO.Q_YEM
TO.Q_GYM

第1条表示接到从松林站发往宣威站的车次为41012。

4.3 电子地图

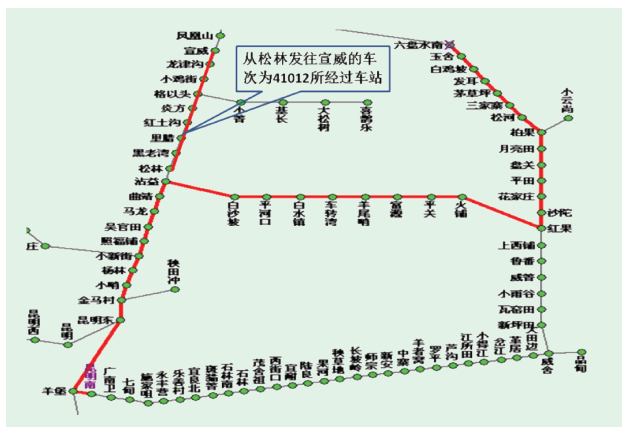


图2 显示列车径路的电子地图

为了让使用者对转报情况有直观的理解,方便转报,利用电子地图功能对整个转报情况进行描述,直观标示出确报列车实际运行轨迹,如图2所示。红线部分反映一个确报从昆明南站出发,经由白水镇,到达六盘水轨迹标示。另外,从图2可以看出,在松林与宣威之间车站所有确报站都进行转发。

5 结束语

系统2010年投入使用至今,运行非常稳定。程序具有对多站转报能力,转报字典通过对车站拓扑图字典做相应维护,不需要对程序本身做任何改动。

责任编辑 杨利明

(上接 P3)

4 数据中心虚拟平台管理

虚拟平台管理人员按层次分为虚拟机管理人员、虚拟群集管理人员和虚拟平台管理人员。虚拟机管理人员负责虚拟机的使用、维护、监控、开关机等。虚拟群集管理人员负责虚拟群集物理服务器、存储、网络的配置及监控、虚拟机的创建、配置、克隆、备份等,其权限包含虚拟机管理人员的权限。虚拟平台管理人员包含虚拟群集管理人员的权限,虚拟群集的创建、配置和监控,虚拟管理平台的安装配置,权限的分配等,主要是铁路局虚拟平台维护人员。

5 结束语

经过3年多的虚拟化技术探索和应用实践,郑州铁路局局级数据中心虚拟化技术应用已经初具规模,目前在铁路局机房,内部网有微机服务器虚拟化平台3个,物理设备(不包括存储)24台,构建虚拟机145个;小型机虚拟化平台1个,物理设备3台,构建虚拟机20多个。外部网微机

服务器虚拟化平台1个,物理设备4台,构建虚拟机19个。在站段有微机服务器虚拟化平台6个,物理设备34台,构建虚拟机182个。虚拟化技术应用已遍布铁路局、站段。数据中心的设备虚拟化,不仅节省了大量的设备采购投资和运营成本,也打破了应用部门、应用项目对物理资源的依赖,用户可以按照约定合理使用资源,不再受资源变更和管理的影响;技术人员可以通过对设备资源状况的检测、监控,实时、灵活地调整资源分配,有效地提高了资源应用效率,为今后提高信息资源共享、消除信息孤岛、提高灾备恢复能力奠定了基础。

参考文献:

- [1] Scott Lowe.Mastering VMware vSphere 5[Z]. Sybex, 2011.
- [2] Implementing IBM Power Virtualization Center in Your Data Center[Z]. IBM Redbooks, 2012.
- [3] IBM PowerVM Virtualization Introduction and Configuration [Z]. IBM Redbooks, 2008.
- [4] Virtual IO Server Integrated Virtualization Manager[Z]. IBM Power Systems, 2008.

责任编辑 陈蓉