

文章编号:1671-251X(2019)08-0054-06

DOI:10.13272/j.issn.1671-251x.2018090042

一种矿用音视频联动软件设计

杜志刚^{1,2}, 陈康^{1,2}, 王伟^{1,2}

(1. 中煤科工集团常州研究院有限公司, 江苏 常州 213015;
2. 天地(常州)自动化股份有限公司, 江苏 常州 213015)



扫码移动阅读

摘要:针对目前矿井融合调度通信系统中音频通信和视频通信缺乏有效联动,无法支撑调度员进行高效音视频联动通信的问题,设计了一种矿用音视频联动软件。软件中间件通过 OCP 协议实时获取调度通信主机终端状态,数据处理模块判断电话状态,并将联动视频信息通过消息队列 RabbitMQ 发送给客户端,客户端接收数据处理模块传送的合法指令,控制视频终端开启和关闭。测试结果表明,该软件运行稳定,安装简便;通过 Web 配置管理平台,用户可以方便地对电话、视频信息进行管理,同时对二者的对应关系进行配置;对每个客户端绑定的调度电话,当该电话主叫或者被叫时,均可在屏幕上显示对端电话绑定的摄像机信息,视频信息显示流畅,可在 1 s 内响应,满足用户的实时性要求。

关键词:煤矿调度通信;音频通信;视频通信;音视频联动;消息队列;多线程;ocp 中间件
中图分类号:TD655 **文献标志码:**A

Design of mine-used audio and video linkage software

DU Zhigang^{1,2}, CHEN Kang^{1,2}, WANG Wei^{1,2}

(1. CCTEG Changzhou Research Institute, Changzhou 213015, China;
2. Tiandi (Changzhou) Automation Co., Ltd., Changzhou 213015, China)

Abstract: In view of the lack of effective linkage between audio communication and video communication in current mine integrated dispatching communication system, it cannot support dispatcher to proceed efficient audio and video linkage communication, a mine-used audio and video linkage software was designed. The software middleware obtains status of terminal of scheduling communication host in real time through OCP protocol, the data processing module judges the phone status, and sends the linked video information to the client through the message queue RabbitMQ. The client receives the legitimate instructions sent by the data processing module and controls the opening and closing of the video terminal. The test results show that the software is stable and easy to install; through the Web configuration management platform, users can easily manage the telephone and video information, and configure the corresponding relationship between the two;for each scheduling phone bounded by client, when the phone is calling party or called, the camera information of phone of the other party can be displayed on the screen. The video information can be displayed smoothly and the response can be made within 1 s, which meets real-time requirements of users.

Key words: coal mine dispatching communication; audio communication; video communication; audio and video linkage; message queue; multithreading; ocp middleware

收稿日期:2018-09-14;修回日期:2019-04-17;责任编辑:张强。
基金项目:天地科技股份有限公司科技创新创业资金专项项目(2018-TD-QN013);天地(常州)自动化股份有限公司技术研究项目(2018GY106)。
作者简介:杜志刚(1986—),男,山东泰安人,工程师,主要研究方向为通信技术、计算机软件设计与开发,E-mail:568850188@qq.com。
引用格式:杜志刚,陈康,王伟.一种矿用音视频联动软件设计[J].工矿自动化,2019,45(8):54-59.
DU Zhigang, CHEN Kang, WANG Wei. Design of mine-used audio and video linkage software[J]. Industry and Mine Automation, 2019,45(8):54-59.

0 引言

煤炭是我国的主要能源,约占一次性能源的70%^[1],煤矿安全生产一直是我国煤炭行业的重点研究课题之一。随着计算机技术的不断发展,“六大系统”^[2]在煤矿的应用不断更新,但是这些系统各自分散。融合通信系统^[3-4]在兼容现有通信系统的前提下,将有线通信系统、无线通信系统、IP广播系统等有效集中起来,涵盖了数据、语音、视频等业务,为跨越各系统之间的通信提供了高效平台和桥梁。但目前绝大部分矿井融合通信系统使用的仍是非可视电话,音频通信和视频通信还缺乏有效联动,无法支撑调度员进行高效音视频联动通信,为此,本文设计了一种矿用音视频联动软件。软件中间件通过 OCP(Operator Communication Protocol)协议^[5]实时获取调度通信主机终端状态,数据处理模块判断电话状态,并将联动视频信息通过消息队列 RabbitMQ^[6]发送给客户端,联动客户端接收服务端发来的指令,分析指令合法性,对有效的指令进行处理,实现对视频终端的打开或关闭。该软件可通过 Web 配置管理界面为电话终端灵活配置不同视频监控仪,为调度指挥、应急救援、故障视频诊断等提供了有力手段和安全保障。

1 软件结构

矿用音视频联动软件是基于 KT379 调度通信系统设计与实现的,系统服务模块运行在 Linux 系统中,各服务模块之间通过消息队列 RabbitMQ 进行消息传递,选择存储速度快、体积小、支持跨平台使用的 MySQL 作为数据存储工具。软件主要包含 4 个功能模块: ocp 中间件、数据处理服务模块 InterForVLC、视频播放客户端及 Web 配置管理平台 AM,结构如图 1 所示。

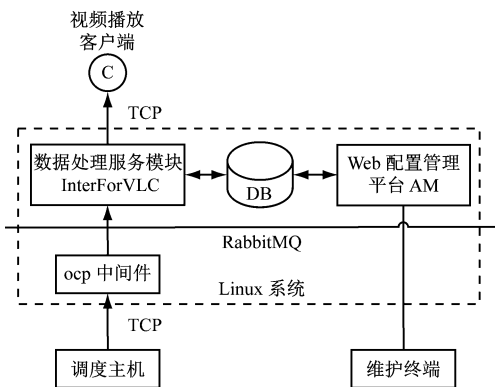


图 1 矿用音视频联动软件结构

Fig. 1 Structure of mine-used audio and video linkage software

1.1 ocp 中间件

调度通信系统中,通信终端信息主要存储于调度主机等交换设备中,无法直接获取。ocp 中间件通过 TCP 协议接收调度主机响应消息,根据调度主机 OCP 协议解析响应消息,从而获取终端配置信息和实时状态信息,并且通过消息队列服务器上报给数据处理服务模块。ocp 中间件工作原理如图 2 所示。

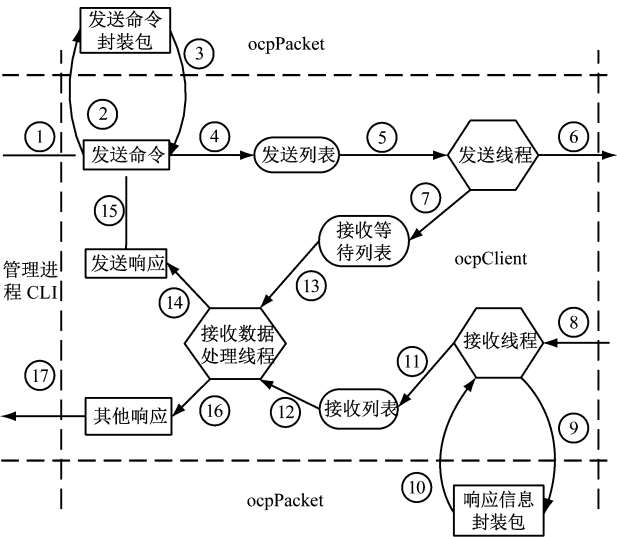


图 2 ocp 中间件工作原理

Fig. 2 Working principle of ocp middleware

ocp 中间件主要包含 3 个并发线程:

- (1) 发送线程:来自管理进程的发送请求经过 ocpPacket 的封装存放于发送列表中,这些数据包括请求数据(需要服务端做出响应)和一般性数据(比如消息回复、修改文件等),发送线程将这些数据按照预定的规则发送至调度主机,并且保存请求数据的唯一性信息(如 ID、序列号等)。
- (2) 接收线程:调度主机对请求做出响应,中间件接收这些响应和其他信息,经过封装存放在接收列表中。
- (3) 接收数据处理线程:提取接收列表中包含的信息,与接收等待列表中保存的序列号等唯一性信息比较,区分这些响应的归属,通过响应消息封装包的封装将响应信息转发给请求者客户端,接收到的其他信息则按照协议做相应处理。

1.2 数据处理服务模块

数据处理服务模块 InterForVLC 初始化运行时,首先创建连接请求处理线程、数据接收线程、数据处理线程和数据发送线程 4 个线程,分别用于接收客户端连接请求、接收 RabbitMQ 数据消息、处理消息和发送消息至客户端。各线程相互独立,同时通过队列进行消息传递,完成对消息的接收、处理和发送。其工作原理如图 3 所示。

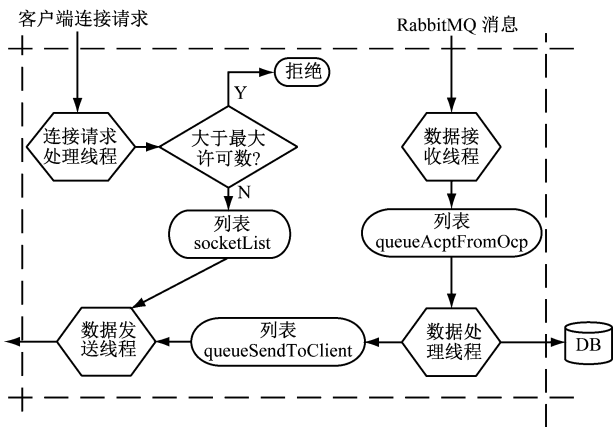


图 3 数据处理服务模块工作原理

Fig. 3 Working principle of data processing service module

(1) 连接请求处理线程: 通过循环执行 ServerSocket.accept() 等待客户端连接请求, 当服务端收到客户端的请求时, 首先通过服务器中的注册文件判断已连接客户端数量是否超过服务器允许的最大数量, 从而确定是否接收此次连接请求, 被允许的连接请求将创建一个新的套接字建立连接, 同时将套接字信息存入连接信息列表 socketList。

(2) 数据接收线程: 作为 RabbitMQ 消费者 (Consumer) 端, 线程首先通过 IP 地址、端口号等信息连接至消息队列服务器, 新建连接的同时建立一个队列 queue, 这样就可以收到消息生产者 (Producer) 通过交换机 Exchange 广播至队列的消息, 接收到的消息存入待处理队列 queueAcptFromOcp。

(3) 数据处理线程: 工作流程如图 4 所示。待处理队列 queueAcptFromOcp 中消息主要有 2 类: 终端配置信息和终端实时状态信息。终端配置信息主要包含终端所属调度主机、标识 ID、终端号码等, 服务端将经过处理后的配置信息存储至数据库, 同时更新内存哈希表; 终端实时状态信息主要包含终端所属调度主机、标识 ID、终端号码、电话状态 ID、对端标识 ID、对端号码等, 当终端状态为呼入振铃 (被叫) 或者呼出听铃 (主叫) 时, 表明终端已经发起呼叫, 需要打开对端视频, 此时从数据库中找到该终端所对应的对端号码绑定的视频信息, 按照协议将整合信息放到待发送队列 queueSendToClient。同理, 当电话状态为正在释放 (主叫和被叫) 时, 添加信息至待发送队列。同时, 接收到的无效数据则直接丢弃。处理失败的操作要及时存入日志处理模块, 对错误地点、错误原因进行日志记录, 以便日后的软件维护。

(4) 数据发送线程: 从待发送队列 queueSendToClient 队首取出一条信息, 循环发送至 socketList 中的各客户端, 各客户端接收到消息

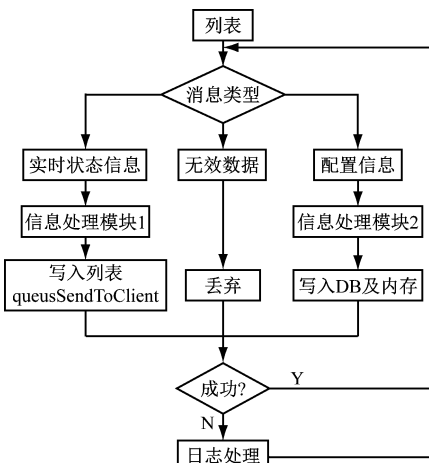


图 4 数据处理线程工作流程

Fig. 4 Work flow of data processing thread

以后, 根据本地所绑定的终端号码信息, 判断消息是否有效, 客户端响应有效消息, 控制视频的打开或关闭。

1.3 视频播放客户端

基于 C/S 架构完成的视频播放客户端主要是接收服务端发来的指令, 分析指令的合法性, 对有效指令进行处理, 实现对视频终端的打开或关闭。客户端与服务端之间通过可靠的 TCP 通信协议进行指令交互, 保证指令及时传达。指令内容采用格式化的字符串形式, 所有字符串采用 UTF-8 编码。一条合法的指令信息格式如下:

```
cmd:video-start|video-stop
seq:4321
class:vlc|zsyh|hk
chan:"192.168.1.13,8000,admin,12345,1"
style:"x=100,y=100,w=400,h=300"
```

指令包含多行, 空行表示指令结束; 每行代表一个信息字段, 由“NAME:VALUE”形式的字符串组成, 一个报文中除结束符 (空行) 在最后外, 其余各信息字段可以乱序。其中, cmd: 指令的类型, 指示客户端打开或者关闭视频; seq: 指令唯一序号 ID, ID 相同表示对同一条指令的二次发送或者响应; class: 视频类型; chan: 视频访问描述信息字符串; style: 视频信息显示窗口的屏幕位置和大小 (其中 x, y 表示窗口起始坐标, w, h 表示窗口大小)。

客户端主要利用视频摄像头厂家提供的 SDK (Software Development Kit) 进行二次开发来实现图像的远程访问^[7], 这里以海康威视视频为例做简要说明, 其主要流程如图 5 所示。调用 SDK 实现实时视频预览、录像回放、云台控制等功能, 实现每个功能时初始化 SDK、用户注册设备、注销设备和释放 SDK 资源是必不可少的。初始化 SDK 是指对整个网络 SDK 系统进行初始化, 完成内存预分配等

操作。用户注册设备实现用户的注册功能,注册成功后,返回的用户 ID 作为其他功能操作的唯一标识。注销设备和释放 SDK 资源是指在功能调用结束或者调用失败后完成内存分配资源的回收。

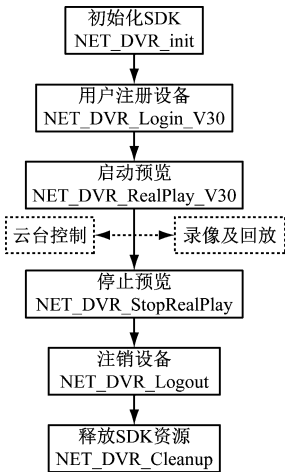


图 5 视频预览调用主要流程

Fig. 5 Main flow of video preview calling

1.4 Web 配置管理平台

Web 配置管理平台主要用来配置和管理音视频关联关系。平台基于 Spring MVC 框架完成,可实现用户界面和业务逻辑分离,灵活的依赖注入降低了应用模块之间的耦合度,各模块既相互独立,又协同合作,同时框架应用可跨各种平台进行部署,具有可扩展性、可复用性、可维护性、灵活性高的优点。选择存储速度快、体积小的 MySQL 作为数据存储工具。MySQL 具有开源、多语言支持、可跨平台使用、可移植性高等优点。

平台前端界面基于 ExtJS 框架开发,服务端 AM 部署在 Linux 系统中,采用 Java 语言与 MySQL 数据库进行数据交互,前后端使用 JSON 进行数据交互。JSON 是一种轻量级的数据交换格式,可在多种语言之间进行数据交换,相比于传统交换模式 XML,JSON 易于阅读和编码,且是 JavaScript 规范的子集,能被支持 JavaScript 的浏览器所解析。目前网络上通用的数据交换格式 XML 在实际应用上存在不少缺陷^[8],JSON 的数据传输效率明显优于其他数据传输格式^[9]。

2 软件实现的关键技术

2.1 稳定可靠的消息传递机制

由于联动软件涉及多个服务模块,各服务模块之间协同合作,为降低模块之间的耦合性,必须寻找一种比较安全可靠、高效快速的消息传递机制。

消息队列是一种应用程序对应用程序的通信方法。应用程序通过读写出入队列的消息(针对应用程序的数据)进行通信。消息传递指的是程序之间

通过在消息中发送数据进行通信,而不是通过直接调用彼此来通信。排队指的是应用程序通过队列进行通信,队列的使用避免了接收和发送应用程序需同时执行的要求。

高级消息队列协议 (Advanced Message Queuing Protocol, AMQP) 是应用层协议的一个开放标准,为面向消息的中间件设计。RabbitMQ 由一个 AMQP 的开源实现。常见的消息交换机 (Exchange) 有 3 种类型^[10-11],即 direct, topic, fanout, 本文选用 fanout 类型进行生产者 (Producer) 端和消费者 (Consumer) 端的消息传递。fanout 就是广播模式,也称之为发布/订阅 (Publish/Subscribe) 模式,可将同一个消息提供到与之绑定的多个消费者队列中。fanout Exchange 工作模型如图 6 所示。

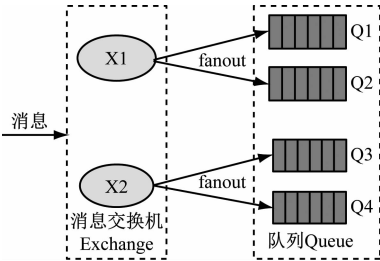


图 6 fanout Exchange 工作模型

Fig. 6 Working model of fanout Exchange

(1) 生产者端:并不会直接发送消息到消费者队列,而是发送到消息交换机 Exchange 中。消息交换机从生产者端接收消息,通过设置交换机类型可以规定如何处理这些消息。

(2) 消费者端:当需要接收新的消息时,将消费者连接到消息队列服务器上,消费者对旧的消息并不感兴趣,因此,新建连接的同时需要一个新的队列。如果在声明队列的时候不指定名字, RabbitMQ 会随机选择一个名字,这样通过新的队列就可以接收到以后生产者广播来的消息,当消费者端关闭连接时,队列也会随之删除。

2.2 OCP 协议

OCP 协议描述了 ocp 中间件与调度主机之间的通信协议,是 TCP/IP 应用层协议,传输层采用 TCP 协议,业务协议端口为 4,中间件协议端口随机绑定,生命周期内保持不变。OCP 协议的消息格式中的消息字段都是网络序的,负责中间件的登录/注销、用户状态的预约、用户状态变更通知等。消息格式如图 7 所示。其中消息长度是指包含消息头和消息体的总长度;事务标识始终在当前中间件与调度主机形成的连接关系中保持唯一;命令字指示消息类型。



图 7 OCP 协议消息格式

Fig. 7 Message format of OCP protocol

ocp 中间件与调度主机进行数据交互时, 先进行登录(0x00000001), 调度主机回复登录成功时, 表示 TCP 连接已经开启, 回复命令中的 expire 表示 TCP 连接生命周期, 连接超时或中断均需重新建立新的连接, 当中间件发送注销命令(0x00000006)时, 调度主机将取消所有预约信息。ocp 中间件登录成功后, 首先向调度主机周期性发送配置数据查询命令(0x00000058), 调度主机响应配置数据查询命令, 将已配置的用户数据回传中间件, 保证配置的用户信息的准确性; 其次发送用户状态预约命令(0x00000007), 调度主机收到预约命令后, 将预约用户实时状态回传给中间件, 同时当预约用户状态变化时, 中间件将收到调度主机发来的带有 user status 的用户状态变更消息。

2.3 多线程应用技术

作为服务端的 ocp 中间件与客户端存在一对多的关系, 大量的并发数据处理、高实时性的数据传递等都要求软件开发要选择先进、成熟、具有强大应用数据处理能力的技术。相比于进程, 运行于一个进程中的多个线程使用相同的地址空间, 共享大部分数据, 启动一个线程所花费的空间远小于启动一个进程, 而且彼此间切换所需要的时间也更少, 是一种花销小、切换快、更“节俭”的多任务操作方式。

Java 中有 2 种实现多线程的方式: 直接继承 Thread 类或者实现 Runnable 接口。比较而言, 选择 Runnable 接口具有以下优点: ① 适合用多个相同代码的线程处理同一资源(如队列), 可实现代码、数据的有效分离。② 避免了 Java 的单继承性带来的局限, 实现了代码优化。③ 代码可被多个线程共享, 与数据是独立的, 有利于程序的健壮性。

2.4 其他技术

一个好的应用软件除了具备用户所需要的功能外, 还应该具备预见并处理可能发生的各种异常的功能。TCP 客户端和服务端已经正常建立连接之后, 其中一方异常关闭, 如何保证两边的数据传输不受影响; 消息队列中, 生产者和消费者如何保证在不影响其他模块正常工作的前提下与服务端断开

之后的重连; 系统捕捉到的异常信息如何处理等诸多问题都是在软件设计时需要考虑的, 如果不能处理好这些细节, 在运行时发生错误则可能导致系统非正常结束, 造成严重的后果。

3 测试验证

为了验证矿用音视频联动软件的安装便捷性、使用可行性、稳定性, 搭建了测试环境, 其网络拓扑结构如图 8 所示。

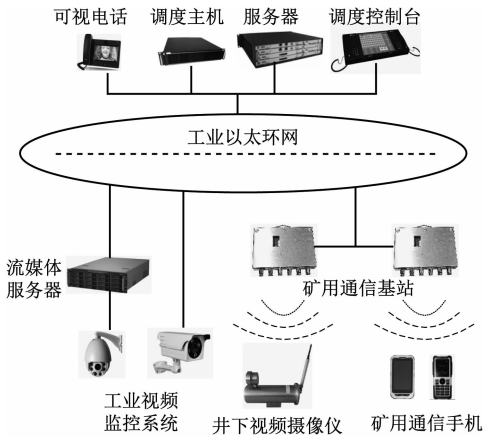


图 8 测试环境网络拓扑结构

Fig. 8 Network topology structure of testing environment

井下视频摄像机通过矿用通信基站接入以太网, 工业视频监控系统中摄像机通过有线方式接入环网, 矿用移动手机终端通过无线方式注册到调度主机中, 其他可视电话、调度电话通过电话线接入通信系统。通过 Web 配置管理平台为每一部电话绑定一台摄像机。测试结果显示:

(1) 音视频联动软件部署在 Linux 系统下, 涉及的所有使用环境、软件通过脚本一键安装, 不同应用场景通过简单修改配置文件即可实现音视频联动功能, 安装简便。

(2) 通过 Web 配置管理平台, 用户可以方便地对电话、视频信息进行管理, 同时可对二者的对应关系进行配置。

(3) 对每个客户端绑定的调度电话, 当该电话主叫或者被叫时, 均可在屏幕上显示对端电话绑定的摄像机信息, 视频信息显示流畅, 可在 1 s 内响应, 满足用户的实时性要求。

(4) 人为进行服务器、客户端主机断线或重启以后, 软件可自启动运行, 功能正常, 日志管理模块可正确、及时记录错误信息; 服务器在长时间运行后, 服务软件运行稳定。

4 结语

设计开发了一套适用于矿井融合调度通信系统

的矿用音视频联动软件,解决了系统实际应用中存在的音频通信和视频通信缺乏有效联动的问题。测试结果表明,该软件能够满足音视频联动通话的需求,安装简便、功能完善、运行稳定。

参考文献(References):

[1] 孙继平,钱晓红. 煤矿事故与应急救援技术装备[J]. 工矿自动化,2016,42(10):1-5.
SUN Jiping, QIAN Xiaohong. Coal mine accident and emergency rescue technology and equipment[J]. Industry and Mine Automation,2016,42(10):1-5.

[2] 国家安全生产监督管理总局,国家煤矿安全监察局. 关于建设完善煤矿井下安全避险“六大系统”的通知(安监总煤装[2010]146号)[EB/OL]. (2010-08-26) [2018-08-13]. <https://wenku.baidu.com/view/a1e50cf4ba0d4a7302763a08.html>.

[3] 顾俊,刘亚兵. 矿井融合调度通信多业务网关系统设计[J]. 工矿自动化,2016,42(11):22-27.
GU Jun, LIU Yabing. Design of multi-service gateway system for mine integration dispatching communication[J]. Industry and Mine Automation, 2016,42(11):22-27.

[4] 薛晓炎,霍振龙,顾俊. 矿井融合调度通信系统的设计及应用[J]. 工矿自动化,2016,42(8):1-5.
XUE Xiaoyan, HUO Zhenlong, GU Jun. Design and application of mine integration dispatching communication system [J]. Industry and Mine Automation,2016,42(8):1-5.

[5] 杨洋. 基于 NC5200A 的融合通信中间件关键技术研究[D]. 南京:东南大学,2012.

[6] 高晓婷. 基于 AMQP 的信息发布与订阅[D]. 杭州:浙江工业大学,2013.

[7] 陈俊良,葛俊峰,叶林,等. 基于海康威视 SDK 的视频监控软件的开发[J]. 工业控制计算机,2015,28(7):97-98.
CHEN Junliang, GE Junfeng, YE Lin, et al. Development of video surveillance software based on Hikvision SDK [J]. Industrial Control Computer, 2015,28(7):97-98.

[8] 罗克,杜志刚,包建军,等. 矿用移动安全管理系统设计[J]. 工矿自动化,2015,41(2):6-9.
LUO Ke, DU Zhigang, BAO Jianjun, et al. Design of mine-used mobile security management system [J]. Industry and Mine Automation,2015,41(2):6-9.

[9] 高静,段会川. JSON 数据传输效率研究[J]. 计算机工程与设计,2011,32(7):2267-2270.
GAO Jing, DUAN Huichuan. Research on data transmission efficiency of JSON [J]. Computer Engineering and Design,2011,32(7):2267-2270.

[10] 李璐,张广泉. 消息中间件的体系结构研究[J]. 苏州大学学报(工科版),2007,27(3):10-14.
LI Lu, ZHANG Guangquan. Research of message-oriented middleware architecture [J]. Journal of Suzhou University (Engineering Science Edition), 2007,27(3):10-14.

[11] 李翠姣. 基于消息队列的分布式系统数据传输优化技术研究[D]. 哈尔滨:哈尔滨工程大学,2015.