基于本体的安防监控指挥系统设备集成及联动控制

郑娅峰,张永强

河南财经政法大学计算机与信息工程学院,郑州 450002

摘要 针对目前在智能安防监控指挥系统中实现设备集成与联动控制的困难,提出了基于本体描述的设备集成方法,设计了新一代安防监控指挥系统的通用体系结构。该结构中首先使用本体描述设备、事件、命令之间的关系,然后定义设备、事件与联动响应动作之间的映射关系,实现面向组合的联动策略设计。在事件响应的核心业务模型中,提出利用多队列缓冲的信号接收技术处理联动设备的事件响应,并给出了具体的响应算法描述。系统有效解决了异构系统(设备)的统一接入问题。实验结果证明,整个系统架构具有较好的可维护性,并在信号的实时处理上具有高可靠性。

关键词 安防监控;设备集成;联动控制;本体

中图分类号 TP393 **文献标识码** A **doi** 10.3981/j.issn.1000-7857.2012.22.008

Device Integration and Linkage Control Based Ontology for Safeguard Control System

ZHENG Yafeng, ZHANG Yongqiang

College of Computer and Information, Henan University of Economics and Law, Zhengzhou 450002, China

Abstract It is difficult to implement the device integration and linkage control in the safeguard control system. In order to do that, this paper proposes a method based on the ontology to integrate devices. A framework for the next generation intelligent control platform of the safeguard system is developed. A core business model is designed for the linkage system in the framework. In this framework, the ontology is used to describe the relationship among the device, the event and the command. A mapping file for the device, the event and the linkage action is build to deal with the linkage strategy. A signal receiver with a multi-queue buffer is provided in the event response model and this technology can handle the incident response of the linkage device. The corresponding algorithms are proposed in the paper. The system solves the unified access to heterogeneous systems (equipment) efficiently. Experimental results show that the entire system architecture has a good maintainability, and a high reliability in the real-time signal processing.

Keywords safeguard control; devices integrated; linkage control; ontology

0 引言

近年来,随着国民经济的发展和人们安全防范意识的提高,我国安防业以每年20%—30%的速度快速成长。安防产品的应用领域越来越广泛,对安防产品的要求也不断提高。目前安防项目的建设已经从不同领域的安防监控走向综合性、智能化安防监控。智能化安防集中监控系统设计以综合布线、电源供电、计算机网络为基础,以视频监控系统为基础系统平台,集成对讲监听、周界控制、门禁等多系统的联动,因此,设备的集成和联动成为整个系统的核心。

然而,由于不同厂家的设备标准不一,有的具有智能控制机制,有的可以通过物理接口进行通信,数据的传输方式也千差万别,这些都造成了设备在集成和联动控制方面的困难。如何有效集成这些异构设备,并使之灵活接入进行联动控制成为安防系统面临的主要问题。目前,在安防集成领域主要存在两大联盟——ONVIF (Open Network Video Interface Forum) 和 PSIA (Physical Security Interoperability Alliance)。

收稿日期:2012-05-09;修回日期:2012-07-10

基金项目:河南省重大科技攻关项目(112102210199)

作者简介:郑娅峰,讲师,研究方向为本体集成、网络通信,电子信箱:zlzyf@126.com

ONVIF 目前以网络视频产品标准化为主,通过和门禁系统之间的互通性,建立全球性的网络接口标准;而 PSIA 致力于使基于 IP 网络的不同安防系统具有兼容性,为实体安防系统的硬件和软件平台创立一种标准化的接口。但两个联盟的标准都属于企业标准,其应用范围存在一定的局限性。

本文提出基于本体的新一代安防监控指挥系统。利用本体描述安防设备,并在此基础上,利用本体具有的良好语义特征,在系统的设备集成、联动控制、事件响应方面提出相应方法,最终构建新一代高性能智能安防监控指挥系统(Next Generation Intelligent Control Platform of Safeguard System, NICPSS)。

1　智能安防系统的体系架构

安防监控指挥系统是事件驱动系统(Event-Based Systems,EBS)的常见应用。而在这类系统的架构设计中,基于分层的架构设计是较为普遍的考虑[1]。分层架构具有较好的分离性和独立性,能够方便地实现分布式开发和部署,易于扩充并可维护。

本文提出了一个典型的安防监控指挥系统分层架构(图1)。整个系统分为4层。最上层是实时监控层,用于实时对系统的运行状况和采集到的信息进行监控,并能够完成对系统的控制操作功能;第二层是业务层,主要提供高度可复用的安防监控服务组件;第三层包括通信层和持久层两个部分,通信层主要提供一个统一的设备信号解析和设备命令驱动总线结构,持久层提供业务层所需的对象持久化服务;最底层是接入层,主要构建一个能够建立在统一设备协议基础上的设备访问服务。

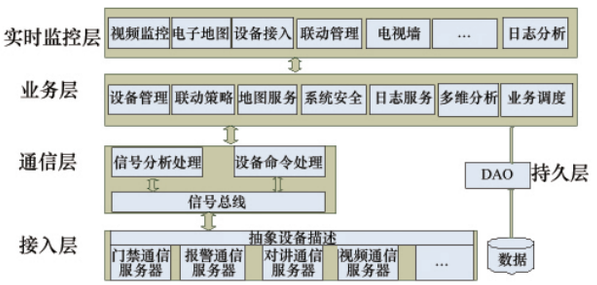

图1　安防监控指挥系统的分层架构图

Fig. 1　Layered architecture for safeguard control system

(1) 接入层

接入层主要包括门禁、报警、对讲、视频等各种通信服务器、相关设备及第三方的业务子系统。由于设备的接入在设备形式、通信方式、通信协议及设备管理层次等诸多方面存在不同,接入层解决的关键是提出一种抽象设备描述机制能够向上屏蔽设备通信的细节,使异构的各类安防子系统以及第三方业务系统能够进行平滑接入。本文使用本体描述对设备进行统一描述以实现设备概念的共享和交互。

在接入层中,存在一个抽象的逻辑设备层(Abstract Device Layer,ADL)。它为整个 NICPSS 提供了统一的抽象设备描述,屏蔽了不同设备的细节,而只考虑和安防有关的信息,并且承担向具体通信系统翻译来自上层的驱动命令。

(2) 通信层

通信层主要包含信号分析处理和设备命令处理两个核心组件。信号的接收和响应是本层的核心。通信层处在接入层之上,它无需考虑底层设备的细节。该层向下面对的是抽象的设备层;上层是业务层,向上提供经过解析后的设备信号和转换业务系统发出的设备控制命令。信号分析处理主要依据不同信号的设备来源、对事件响应的影响等对信号进行优先级处理。设备命令处理主要支持设备间的联动策略配置,当特定事件发生时,能够自动触发该事件和设备所关联的联动策略。

(3) 业务层

业务层处在通信层和数据访问层之上,其核心组件包括设备管理、联动策略、地图服务、系统安全、业务处理等。设备管理可以对所有安防设备及其状态进行统一管理。联动策略根据建立的联动业务规则模型,对信号源、信号类型等触发业务调度操作;系统安全对系统范围内的事件,如系统事件、设备事件、联动事件等进行记录,满足针对系统运行的行为分析。

(4) 实时监控层

建立在业务层之上的实时监控层提供了人机交互界面,面向监控人员主要提供以下功能组件:视频监控实时呈现监控所有设备的状态变化,并对特定状态的出现提供即时告警服务;电子地图能够通过多种交互方式,如命令、地图操作等,定位系统范围内的任意设备,查看某一设备采集的监控信息,如调度视频监控系统某一摄像头对特定区域进行监控;联动管理能够依据信号类型、信号源、时间等进行联动业务规则的设置;日志分析基于系统的日志记录,利用成熟的统计分析方法和人工智能技术,对安防系统平台记录的信号间存在的可能关联,如时间、设备、人员等进行智能分析,对可能的异常进行报警。

2　关键技术的实现

统一的设备接入管理、联动策略定义、事件响应以及自身的智能安全防范感知是实现 NICPSS 系统的关键。

2.1　基于本体的安防系统设备描述

本体(Ontology)通过对以特定领域的概念、术语及其相互关系的规范化描述,勾画出该领域的基本知识体系和描述语言。作为一种说明机制,本体通过不同的应用增强了知识共享和重用[2-3]。

目前在安防领域,尚未有权威的本体构建方法,但针对设备本体的构建方面的经验较多[4-8]。在安防领域,不同的安防子系统是由物理上相关联的若干设备构成的,在应用上考察安防系统主要关心的是设备间的连接控制关系、可以触发的事件集合以及驱动系统工作的命令集合。除了传统意义上的

安防设备,当前安防集成平台发展的一个趋势就是也能够集成第三方的业务系统。例如,在医疗监护领域能够和住院管理系统相连接。一个第三方的业务系统同样可以抽象为消息和命令(不同的服务调用)。基于此,将 NICPSS 系统可以定义为一个 4 元组。

定义 1:$S=(D,E,C,<D)$。其中 D 表示安防子系统的设备集合,$D=\{D_i|0<i\leqslant n\}$;E 表示系统可以触发的事件集合,$E=\{E_i|0<i\leqslant n\}$;C 表示系统可以执行的命令集合,$C=\{C_i|0<i\leqslant n\}$;$<D$ 表示 D 上的偏序关系,表示设备间的连接控制关系。

将 NICPSS 系统中的安防设备定义为一个 5 元组。

定义 2:$DO=(A,E,C,G,T)$。其中,A 表示设备属性的集合,$A=\{A_i|0<i\leqslant n\}$;E 表示设备可以触发的事件集合,$E=\{E_i|0<i\leqslant n\}$;C 表示设备可以执行的命令集合,$C=\{C_i|0<i\leqslant n\}$;G 表示该设备和其他设备组成的逻辑控制组集合,$G=\{D_i|0<i\leqslant n\}$;T 表示一个设备和同子系统内其他设备之间的物理拓扑结构,$T=\{(a,b)|(a\in D,b\in D)\}$,当 $a=b$ 时,表示系统是由若干独立的设备节点构成的。

依据安防子系统、设备、关系、事件及命令等概念,利用 UML 建立针对 NICPSS 的本体描述模型,如图 2 所示。

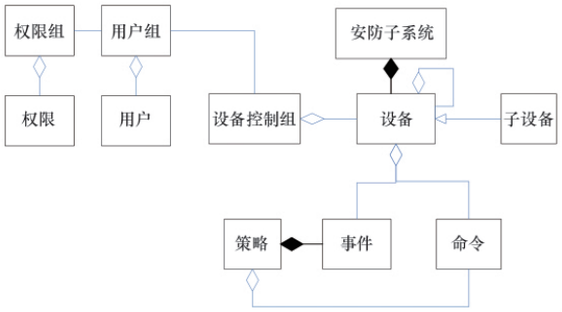

图 2　安防系统的本体模型

Fig. 2　Ontology model of safeguard system

在图 2 中,设备和子设备属于“Kind-of”关系。在系统模型中,设备概念属于一种抽象的设备描述,子设备则属于根据需要扩展具体特性的设备定义。而抽象的设备概念是由其自身的属性(位置、编号等)、围绕该设备触发的事件以及驱动设备工作的命令集构成。它们之间的关系属于“Part-of”。类似地,一类事件触发一种策略,而每个策略则会发出若干个命令,因此,策略和事件、命令之间也构成了整体和部分(Part-of)关系。一个设备控制组属于在操作上相关联的设备组合,它们之间的关系也属于 Part-of 关系。

2.2　面向组合的联动策略设计

联动管理的基础是统一的设备管理。每个设备在系统中都必须能够唯一地表示,以便系统能够精确地对设备进行管理。安防系统的设备从物理形态上可以分为两类:基于固件的安防设备和以纯软件形态表现的业务系统。

通常的安防设备可以通过 RFC、TCP/IP、串口通信等方式进行交互,获得设备发出的各类报警信号,并通过对应的方式将响应转化为设备命令驱动设备工作;对于第三方的业务系统,更多的是通过 RFC 方式获得系统的状态或驱动系统,某些封闭的系统甚至需要通过数据库访问的形式来进行集成。在 NICPSS 模型中,将来自外围设备的信息统一定义为事件(Event),描述一个事件的主要属性包括时间、事件类型、事件源(系统+设备)、优先级等。而将驱动外围系统工作的消息明确为命令原语(Command),命令原语的关键属性包括时间、命令、目标(系统+设备)、优先级等。

联动管理的核心是策略的管理,每个策略均描述了设备、事件与联动响应动作之间的映射关系。当某个设备相应事件发生时,不同的设备执行相对应的联动响应动作。图 3 描述了联动子系统的核心业务模型定义。

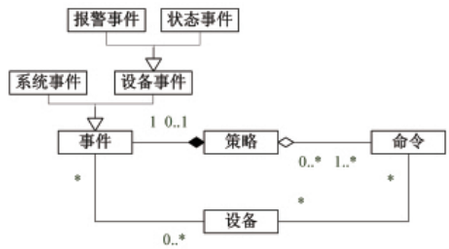

图 3　联动子系统的核心业务模型

Fig. 3　Core business model of linkage system

一个策略可以定义为二元组。

定义 3:$T=(E_i,C)$。其中,E_i 表示触发策略执行的一个事件;C 表示该策略执行时的命令集合,每个动作对应一个系统的命令(原子服务),这个命令可以表示为

$$C=a(x_1,\cdots,x_n)S(P,R)$$

其中,a 为原子动作名;P 和 R 分别为动作的前提条件和执行结果;$x_1,\cdots,x_n$ 为在 P 和 R 中出现的所有个体名[9]。

在联动响应中,当一个门禁打开的信号到达时,最简单的联动策略是调取该门禁附近的摄像头采集的视频信息到显示屏,接下来的联动策略是将打开门禁的出卡人信息显示在操作台上。下面是一个简单的门禁打开的联动策略的算法描述。

```
begin
door=从门禁信号中获得设备信息;
cardholder=从门禁信号中获得持卡人信息;
if 非法卡 then
   创建非法闯入报警信号;
   return;
else
   获得对应的联动策略;
   actions=从策略中获得命令集合;
   for action in actions
     begin
       执行当前的 action,发送命令到命令队列缓冲区;
     end
end
```

在上述策略中,调取对应的摄像头一般需要调用视频服务的函数,持卡人的信息可能来自第三方的系统调用返回的信息,通常采用 SOAP、EJB 等。因为服务形式的易变性,构建统一的命令调度变得非常复杂。文献[9]提出了基于动态描述逻辑的 Web 服务自动组合框架。在 NICPSS 中,由于设备的接入被封闭在接入层,有专门的通信程序接收来自上层的命令驱动,因此,适合将所有的服务定义为原子服务,而原子服务的翻译由接口程序负责。

2.3　事件响应模型

NICPSS 系统的核心服务首先是能够对系统的各类事件作出反应。从图 3 可以看出,信号的类型分为系统信号和设备信号。系统信号来自系统本身,体现了系统当前运行状态,如稳定状态、安全状态等;设备信号来自不同的安防子系统,如周界系统的状态信号、某个防区的报警信号等。

根据前述的要求,NICPSS 设计了一个多队列缓冲的信号接收处理系统,图 4 为该处理系统的基本模型。

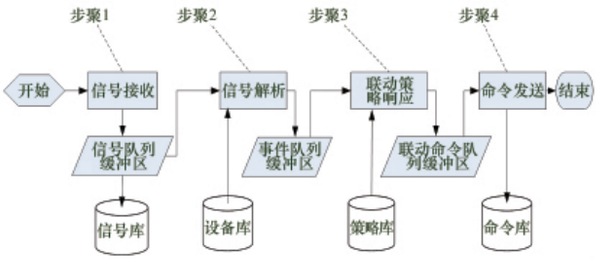

图 4　事件处理模块结构图

Fig. 4　Event process model

步骤 1,信号接收组件不间断地接收来自不同安防子系统的设备信号,并经过简单拆包后送入双向存取的信号队列缓冲区,同时触发信号解析组件;

步骤 2,信号解析组件从信号队列缓冲区读取信号数据,根据预读取的设备描述信息解析设备信号,产生设备事件送入事件队列缓冲区,并触发联动策略;

步骤 3,被触发唤醒的联动策略处理器从事件队列缓冲区读取设备信号,并根据预定义的联动策略,创建设备驱动命令,并存入联动命令队列缓冲区,同时触发命令发送组件;

步骤 4,被唤醒的命令发送组件从命令发送队列读取命令,创建命令的 UDP,发送。

图 5 是这个联动子系统的基本类图。其中关键是 Signal 类的实现。其基本结构如下。

```
import java.io.Serializable;
import java.util.Observable;
public abstract class Signal
                extends Observable implements
Serializable {
    //心跳信号
    public static final String
SIGNAL_HEARTBEAT="HeartBeat";
```

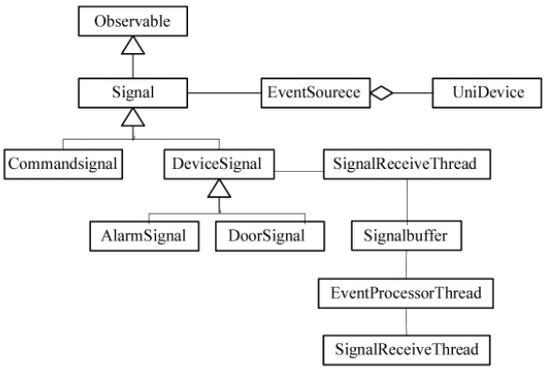

图 5　联动子系统的类图

Fig. 5　Class diagram of linkage system

```
    //对应于产生该信号的子系统代码
    public String subSysNo;
    public Signal() {
        super();
    }
    /* 执行该方法,触发对应的联动 */
    public void send() {
        this.setChanged();
        this.notifyObservers(this);
     }
    }
```

信号解析线程和信号接收线程共享一个信号缓冲队列,该队列的访问必须实现多线程的对象互斥访问机制。

```
public class SignalParser  implements Runnable  {
    private LinkedList<DeviceSiganl> signalPool
                        =new
LinkedList<DeviceSiganl>();
    private boolean workingStatus;
    public SignalParser (
                LinkedList<DeviceSiganl>
signalPool){
        super();
      this.signalPool=signalPool;
      workingStatus=true;
    }
    public DeviceSiganl pickFromSignalPool(){
      DeviceSiganl signal=null;
        synchronized(signalPool){
            while(signalPool.isEmpty()){
                try {
                    signalPool.wait();
                } catch (InterruptedException e) {
                    //省略异常处理
                }
```

```
                }
            signal=signalPool.poll();
                }
        return signal;
        }

        public void run(){
                while(workingStatus){
                        DeviceSignal
signal=pickFromSignalPool();
                        //处理信号
                }
        }
}
```

3　测试及分析

系统运行可靠性和性能是系统用户特别关注的两个指标。针对本项目的特点,重点测试信号响应时间和遗漏率两项指标。安防系统发出的信号有着不同的优先级,其响应时间和响应质量也有不同的要求,系统应确保高优先级的信号无论在任何情况下都能及时得到响应。

因此在具体设计时,将每种安防系统的信号类型分为普通和紧急两种优先级别。其中紧急信号要求任何情况下都不能出现遗漏,在测试用例的数量比例上遵循 5:95 的模式对两种信号进行分配。在极端模式下,根据不同子系统的特点进行特别设计,例如视频监控信号的发送在极端模式下的峰值比正常模式增大了 10 倍,而门禁和对讲则按照 100%的峰值设计了紧急模式下的信号产生规模;另外,不同安防子系统的信号产生频律也是不一样的,在实际中对于系统的服务质量也有一定的影响。表 1 列出了 4 个安防子系统的组合测试设计,分别定义了在正常模式下和极端模式下的测试设计。

表 1　测试设计方案

Table 1　Scheme of design and test

安防子系统	监测点数量	正常模式		极端模式	
		普通信号	紧急信号	普通信号	紧急信号
视频监控	1120	2240	112	2240	1120
周界报警	93	186	9	186	93
门禁	242	242	12	242	242
对讲	528	528	26	528	528

基于上述设计,选择 PowerEdge 2950 服务器(1CPU,4GB 内存,千兆以太网)作为测试服务器进行测试。表 2 列出了不同组合下信号的最长响应时间和处理率的测试结果。

从表 2 可以看出,在正常模式下,响应时间和处理率都达到了较好的应用要求。在极端模式下,紧急信号的最长响应时间有一定的上升,达到了 0.27s,但仍小于系统规定的 0.5s 的最长响应时间。普通信号的最大响应时间达到了 0.4s,而且出现了 0.8%的遗漏率。主要原因在于普通信号包含了大量的状态信号,而状态信号和时间密切相关,重复的状态信号导致了大量无效的状态检测操作,增加了系统处理时间,导致少量信号长期处于栈底无法得到处理。下一步的研究可以在此方面进行改进。

表 2　测试结果

Table 2　Test results

测试项目	最长响应时间/s		处理率/%	
	普通信号	紧急信号	普通信号	紧急信号
正常模式	0.15	0.15	100	100
极端模式	0.4	0.27	99.2	100

4　结论

智能安防监控指挥系统的主要特点是设备集成、联动控制和智能安全防范。为了解决安防监控指挥系统在设备集成及联动控制方面面临的诸多问题,本文利用本体技术构建共享的安防设备知识模型,为新一代智能安防监控指挥平台提出了一个通用体系结构 NICPSS。论述了 NICPSS 实现设备管理、联动控制及事件响应的方法。作为高性能的联动指挥系统,NICPSS 在应用中得到了认可。下一步在安防领域的本体模型的通用性和开放性方面还需要做进一步的细化。未来的研究将关注如何利用面向高维数据的情景感知技术提升 NICPSS 在安全防范感知方面的能力。

参考文献(References)

[1] Voisard A, Ziekow H. ARCHITECT: A layered framework for classifying technologies of event-basedsystems[J]. *Information Systems*, 2011, 36(6): 937-957.

[2] 王楠, 欧阳丹彤, 孙善武. 基于本体的分层抽象模型 [J]. 计算机科学, 2011, 38(2): 184-186.
Wang Nan, Ouyang Dantong, Sun Shanwu. *Computer Science*, 2011, 38 (2): 184-186.

[3] Neches R, Fikes R, Finin T, *et al*. Enabling technology for know ledge sharing[J]. *AI Magazine*, 1991, 12(3): 36-56.

[4] Lee J H, Suh H W. OWL-based product ontology architecture and presentation for sharing product knowledge on a web [C]//Prceedings of 27th Computers and Information in Engineering ConferenceL Vegas, NV, USA: ASME, 2007: 853-861.

[5] Zhan P, Jayaram U, Kim O, *et al*. Knowledge representation and ontology mapping methods for product data in engineering applications [J]. *Computing and Information Science in Engineering*, 2010, 6(10): 1-10.

[6] Chen Y J. Development of a method for ontology-based empirical knowledge representation and reasoning [J]. *Decision Support Systems*, 2010, 50(1): 1-20.

[7] Moon S K, Kumara S R T, Simpson T W. Knowledge representation for product design using techspecs concept ontology [C]// Information Reuse and Integration, Conf, Washington, DC: IEEE System, 2005: 241-246.

[8] 胡玉杰, 李善平, 郭鸣. 基于本体的产品知识表达[J]. 计算机辅助设计与图形学学报, 2003, 15(12): 1532-1537.
Hu Yujie, Li Shanping, Guo Ming. *Journal of Computer-Aided Design Computer Graphics*, 2003, 15(12): 1532-1537.

[9] 万长林, 韩旭, 牛温佳, 等. 基于动态描述逻辑的服务组合及质量模型[J]. 电子学报, 2010, 38(8): 1923-1928.
Wan Changlin, Han Xu, Niu Wenjia, *et al*. *Acta Electronica Sinica*, 2010, 38(8): 1923-1928.

(责任编辑　安莹,吴晓丽)