

# 传感器网络操作系统问题与进展<sup>\*</sup>

石高涛 廖明宏

(哈尔滨工业大学计算机科学与技术学院 哈尔滨 150001)

**摘要** 传感器网络集成了传感器、微机电系统和网络等技术,提供了一个全新的分布式信息搜集和处理平台。作为上层协议和应用程序的运行基础的操作系统,传统的嵌入式操作系统已难以适应传感器网络的要求。从操作系统的角度来分析传感器网络的特点,给出了传感器网络操作系统亟需解决的关键技术,综述了传感器网络操作系统的当前研究状况,并指出其发展趋势和方向。

**关键词** 传感器网络,操作系统,嵌入式系统

## A Survey on Sensor Networks Operating Systems

SHI Gao-Tao LIAO Ming-Hong

(School of Computer Science & Technology, Harbin Institute of Technology, Harbin 150001)

**Abstract** Sensor networks, which is integrated with sensor, micro-electro-mechanism system and networks technology, provide a novel distributed platform for gathering and processing information. Sensor networks operating system (SNOS) is the basic for running the high-level protocols and applications. In this paper, the characteristics of sensor networks are analyzed from the point of view of OS and the critical technique to be settled in SNOS are provided. Then the exiting SNOSs are surveyed and the trend and direction are also provided.

**Keywords** Sensor networks, Operating system, Embedded system

无线传感器网络是一种新兴的技术,随着普适计算<sup>[1]</sup>研究的深入,越来越多地受到人们的重视。无线传感器网络是由大量形体较小、能源受限并且配置有计算能力、存储能力和无线通信能力的传感器节点组成的无结构网络<sup>[2]</sup>,集成了传感器、微机电系统和网络等技术。节点之间能够协作地实时监测、感知和采集网络分布区域内的各种环境或监测对象的信息,并能对这些信息进行处理,获得详尽而准确的信息,传送到需要这些信息的远端用户,因此,传感器网络旨在提供一个有效的连接自然界和计算机世界的桥梁,可广泛应用于军用、民用及科技领域,如敌情探测、目标跟踪、医疗健康、智能建筑、环境科学和空间探索<sup>[2,3]</sup>等等。

尽管无线传感器网络为我们描绘了一个美好前景,但是由于网络中节点数量多并具有高度自主性,每个传感器节点的能力和能源都十分有限,并且网络工作环境恶劣,网络拓扑始终处于高度的动态变化中,因此构建一个有效的传感器网络系统仍然是一项具有挑战性的工作,特别是在体系结构、网络协议以及系统软件方面,现有的思想与方法完全不适应或者必须经过修改才能应用于传感器网络。

传感器网络操作系统,作为其它上层应用和协议的运行基础,运行在每个节点上,负责管理节点和网络中的软硬资源,因此成为影响系统成败的重要因素。理想情况下,用户应当能够使用操作系统提供的接口直接开发应用程序,而不用过多地关心底层硬件和节点基本功能。

本文的目的是研究传感器网络在操作系统方面带来的挑战性问题。首先从操作系统的角度来分析传感器网络的特

点,找出影响操作系统设计的关键因素,进而给出传感器网络操作系统亟需解决的关键技术,最后分析现有的操作系统优缺点并给出发展趋势。就我们所知,目前还没有有关传感器网络操作系统方面的综述。

## 1 传感器网络特点

### 1.1 传感器网络体系结构

传感器网络是由大量传感器节点组成的,其中的每一个传感器节点都是由电源、感知部件、嵌入式处理器、存储器 and 通信部件这几部分构成<sup>[3]</sup>,电源为传感器提供正常工作所必需的能源,感知部件用于感知、获取外界的信息,并将其转换为数字信号,处理部件负责协调节点各部分的工作,如对感知部件获取的信息进行必要的处理、保存,控制感知部件和电源的工作模式等,通信部件负责与其他传感器或观察者的通信。此外,每个节点上还有运行在其上的软件,为传感器提供必要的软件支持,如嵌入式操作系统、嵌入式数据库系统等。

传感器网络中的节点可以通过人工埋放、飞行器散播,甚至可以借助火箭弹射等方式配置到观测区域。之后,这些节点以自组织的方式构成网络,并通过多跳转发的方式将感知到的数据传输到远端的基站节点(即 sink 节点),而 sink 节点则可以将整个网络的数据通过卫星传输远程中心,同时也可以将用户的查询请求发送到传感器网络中。在这种网络体系结构中,每个节点既可以是感知节点,具有感知环境信息的功能,也可以是路由节点,将接收到的数据向 sink 节点转发, sink 节点充当了远程用户和传感器网络之间的桥梁。因此,

<sup>\*</sup> 基金项目:哈尔滨工业大学校基金(HIT 2002. 74)。石高涛 博士研究生,主要研究领域无线传感器网络,网络嵌入式系统;廖明宏 博士,博士生导师,主要研究领域为新型操作系统,实时与嵌入式计算,对等计算。

整个网络体系结构如图 1 所示。

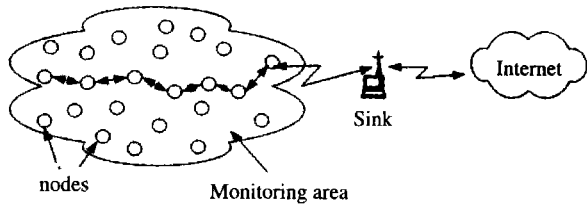


图 1 传感器网络体系结构

此外,由于传感器网络是和应用密切相关的,因此,不同的应用,传感器网络体系结构会有所变化。比如分布在观测区域的传感器可能会移动,有些应用需要多个 sink 节点,每个 sink 节点也可能移动<sup>[4,24]</sup>等。

## 1.2 传感器网络特点

从上面的介绍我们可以看到,传感器网络是一种新型的分布式计算平台,具有完全不同于传统网络的新型体系结构,并且由于传感器节点特点及应用的要求不同,给软件的设计带来了很大的挑战。从操作系统的角度来看,传感器网络的以下特点影响着软件的设计。

1.2.1 资源有限 传感器网络是由大量小型号的节点组成,出于经济上的考虑,这些节点不能配置高档部件,因此导致每个节点上的资源极其有限,主要表现在以下几个方面:

(1)计算能力有限。传感器节点是一个智能的实体,一般都具有嵌入式处理器和存储器,具有一定的计算能力,可以完成一些信息处理工作。但是,由于嵌入式处理器和存储器的能力和容量有限,传感器的计算能力十分有限。目前的传感器节点的普遍配置是内存在几十 k 左右,CPU 频率在 1000MHZ 左右。

(2)电源能量有限。传感器节点是通过电池供电,由于节点散播出去之后就处于无人看管的状态,因此这些传感器节点的能源极其有限并不可再充,节点的电源一旦用尽就无法继续工作。能源有限是传感器网络区别于其它传统网络(包括 ad hoc 网络)的重要特点。

(3)通信能力有限。传感器节点也配置有无线收发装置,但是其通信带宽窄而且经常变化,通信覆盖范围只有几十到几百米。另外,由于邻居节点之间的信号干扰,经常导致通信失败,因此通信也不可靠。

1.2.2 工作环境恶劣 传统的网络主要用于信息的传输与共享,节点一般情况下有人管理,而传感器网络是用来感知环境信息的,因此它不可避免地要散布到将要观测的区域中去,如配置到森林区监测温度、配置到战场观测敌情等。这意味着节点随时都会因为风雨雷电等自然环境的影响而导致节点失效,并且其失效频率也比传统网络高得多。同时,这些节点也会被恶意攻击者拿去,从而窃取其中的有用数据。因此,与传统网络相比,传感器网络的工作环境极其不可靠。

1.2.3 高度动态性 传统网络一般都有固定的路由器、域名服务器等,而传感器网络中不存在这些预先定义好的基础设施,网络拓扑时刻变化,主要表现在以下几点:

(1)它是一种无结构的网络,所有节点都可以充当路由角色。网络配置前,不存在节点绑定到一个固定路由器上的情况,所有通信都必须根据即时的网络情况决定。这一点类似于无线 ad hoc 网络。

(2)如同 1.2.2 所述,节点随时都有可能因为能源耗尽或者自然因素而无法正常工作。

(3)节点有可能不停地移动,比如在鸟类习性观测<sup>[37]</sup>中,每个节点附在鸟的身上,随时随着鸟的移动而改变其位置。

(4)为了最大限度地延长网络寿命,很多节能算法<sup>[32~35]</sup>要求节点根据网络拓扑情况周期性地睡眠,这些处于睡眠状态的节点将无法完成正常的工作。

1.2.4 以数据为中心 传感器网络的使用场景以及和用户的交互方式非常独特,用户和节点之间不仅仅只存在着数据交换,用户很少关心数据是从哪个节点读出来的,在用户眼中,他是在和整个网络进行交互而非某个或某几个节点。这是和传统网络以主机为中心的交互方式的重要区别,在传感器网络中称为以数据为中心。因此,目的地的编址不是通过节点标识,而是依赖于信息的内容属性<sup>[36]</sup>。为了满足这一点,传感器网络必须能够形成一个与应用相关的分布式系统来提供用户请求的数据。

1.2.5 上下文感知 传感器网络用来观测和感知周围环境信息,如温度、湿度、声音、速度和位置、时间等等,这些上下文信息可以用来定义和标识自然界的对象。传感器网络是能够感知、计算和触发的分布式系统,这在传统计算机系统中是不可能出现的。传感器网络操作系统或者其编程环境应当能够在统一框架中考虑这些信息,从而达到计算世界和自然界的无缝结合。

## 2 操作系统关键技术

在传统台式电脑或者手持式设备上,操作系统从功能和体系方面都已近于完善,而对于传感器网络操作系统,其研究才刚刚开始。由于其基础结构和应用的独特性,传统的嵌入式操作系统难以适应新的要求,操作系统的设计存在很多挑战性的问题。这一节详细讨论有关传感器网络操作系统需要解决的关键技术。

### 2.1 体系结构

传感器网络是一个新型的体系结构,其中的每一个节点上的资源都极其有限,并且节点需要时刻监视环境信息的变化,并把发生的事件发送给用户,传统嵌入式操作系统无法很好地满足这些需求,主要原因是传统的嵌入式操作系统是面向实时控制的,而传感器网络需要和周围环境交互,并且能够以自治的方式进行主动计算(Proactive computing)。因此,传感器网络操作系统应当具有什么样的体系结构才能更好地适应于传感器网络是一个仍未解决的问题。

由于传感器网络节点资源有限,因此系统和应用程序的代码量小是其必须的要求。基于组件的体系结构能够使应用程序只包含必要的部分,而用不到的则无须包含进来,因此可以减少应用对内存的需求。TinyOS<sup>[5]</sup>就是采用了基于组件的结构。此外,由于传感器网络需要感知事件信息,对于发生的事件随时做出反应,因此事件驱动模型是操作系统结构的另一选择。事件驱动还能够有效减少能源消耗,因为在没有事件发生时,一些操作系统部件可以休眠,当事件发生时再将其唤醒。如 Tinyos、SenOS<sup>[6]</sup>、MANTIS<sup>[7]</sup>等操作系统都引入了事件驱动模型。事件驱动的另一含义是系统根据事件类型,将自身状态从当前的状态转换到另一个状态,因此基于状态机的模型非常符合这一特点,SenOS 就是基于有限状态机模型的操作系统。

传感器操作系统是一个分布式的系统,但是目前的大部分工作仍然集中在单节点操作系统而忽视了传感器网络的整体性和协作性,传感器网络的以数据为中心的特点也从另一

方面强调了系统的协作性。此外,由于节点资源有限,传统的中间件与操作系统相分离的方法已不适用于传感器网络。因此,有必要开发一个分布式的操作系统,将中间件和底层操作系统融合在一起,为应用程序员提供单一系统映像,屏蔽具体的网络细节。系统应协调应用程序适应网络的状态,从而满足多样的需求。MagnetOS<sup>[5]</sup>采用了虚拟机的思想提供了一个分布式的环境,应用程序可以自动分解成对象并且能够在网络中移动。由于传感器网络的资源有限,任务的分布式协作可能带来大量的消息交换,因此在资源受限的环境下提供一个性能良好、服务丰富的分布式操作系统不是一件容易的工作。

## 2.2 资源管理

传感器网络操作系统仍然不失传统操作系统的功能,因此它也负责管理系统中的软硬件资源。传感器网络中的资源包含节点上的资源(如计算资源、存储资源和电源等)和网络的资源(如网络带宽和服务)。目前的嵌入式技术对管理计算和存储资源的研究已经很成熟,而对于其它资源如电源、网络带宽的管理在传感器网络中还有很多问题需要解决。

**电能管理** 电源是网络中最重要的资源,它决定着网络的寿命。对电源的管理,一方面需要和其它资源协调,“节约”地使用。有研究表明,传感器传输1位信息所需要的电能足以执行3000条计算指令,因此可以通过数据融合<sup>[9,29]</sup>的方法减少数据的传输。假设传输长度为 $S$ 的数据包需要能量为 $E$ ,将数据包融合为长度 $M$ 后传输需要能量为 $e$ ,其中融合所涉及的计算消耗能量为 $a$ ,一般来说 $a+e < E$ 。另一方面,节点应当尽量避免空转或者idle-listening。因为传感器网络主要是感应环境事件变化,而事件的发生是离散的,如果节点时刻处于“警惕”状态,势必造成不必要的能源浪费。

为了避免节点idle-listening,节点需要周期或非周期地睡眠。由于传感器网络存在着大量的节点,邻居节点之间感知范围互相重叠,因此可以在保证网络连通的情况下,使某一区域只保留一个节点(leader or cluster)工作,其它节点睡眠,并且节点之间轮流担当工作节点。假设某一区域有 $n$ 个节点,每个节点的平均寿命为 $L$ ,如果所有节点同时处于工作状态,则网络寿命最多为 $L$ ,而如果节点轮流充当工作节点,则网络寿命会延长到 $n * L$ 。理想情况下,节点不应该一直工作到失效<sup>[12]</sup>,而是以某个时间片为周期交替睡眠,从而保证节点同时失效。因此,节点应以 $t < L$ 为周期工作<sup>[10]</sup>。

另外一种方法是预测事件发生频率,根据频率决定是否睡眠和睡眠的时间长度。如文[11]中假设事件产生符合泊松分布,由此计算出事件发生的概率,根据节点记录的事件发生频率来决定节点是否睡眠。但是节点在不同的状态之间转换也要耗费一定的能源,如果节点睡眠时间不恰当,节点频繁在两个状态之间转换,会无谓地消耗一些能源,一个极端情况是节点睡眠时间为0,那么在转换过程中所消耗的能量完全是净能量损耗,因此只有睡眠时间大于一个极限值 $T_{n,k}$ 时,才会有所收益。文[11]给出了这个极限值,其中 $P_0$ 代表节点处于完全工作状态的单位时间耗能量, $P_k$ 代表接点处在 $K$ 级睡眠状态的单位时间耗能量, $\tau_{d,k}$ 和 $\tau_{u,k}$ 分别代表接点从活动状态转到睡眠状态和相反方向转换需要的时间。

$$T_{n,k} = \frac{1}{2} [\tau_{d,k} + (\frac{P_0 + P_k}{P_0 - P_k}) \tau_{u,k}]$$

**网络带宽管理** 传感器网络无线通信带宽非常有限,单个节点的数据产生能力远远超过可用的网络带宽,如果不在

节点之间协调管理数据感知频率,将无法最优地分配网络带宽。另外,带宽的分配应当依赖于任务重要性和紧急程度,因此操作系统应当能够智能地管理有限的带宽资源。

## 2.3 编程模式

将来的传感器网络无疑会由成千上万个节点组成,对传感器网络中的每一个节点单独编程是一件非常麻烦而又容易出错的事情,传统的编程方法使用底层编程语言、依赖于基本的操作系统抽象、直接和硬件及网络交互已不能满足要求。更重要的是,传感器网络被用来管理各种类型的环境现象,它和自然环境无缝结合并且紧密交互,目前的传统分布式系统编程模式如COBA<sup>[27]</sup>,RPC<sup>[28]</sup>等不能自然地表达这一关系,因此亟需一种编程模式来适合传感器网络编程。

编程模式应当能够根据高层感知任务的说明,自动对单个节点进行编程。程序员不应当过多地关注底层的抽象和诸如节点之间通信连接的创建和撤销等功能。理想情况下,编程环境将传感器网络看成一个虚拟的整体,提供给用户能够抽象定义全局任务的高层概念视图,而底层系统将它们转换成单个节点上的计算和通讯活动。

目前有三种编程模式:数据库方法、基于代理的方法和虚拟机的方法。TinyDB<sup>[9]</sup>和COUGAR<sup>[13]</sup>将传感器网络看成一个分布式的数据库,用户使用类SQL查询语句和整个网络交互,从而减轻与底层API交互的负担,其缺点是不灵活,SQL无法表达复杂的任务。SensorWare<sup>[14]</sup>采用移动代理的模式,每个节点上运行着轻量级的脚本运行环境,注入到网络中的程序以脚本语言写成,并且能够根据事件的移动复制到其它节点。这种方法负载过大,不适合能源受限的传感器网络。MagnetOS提供了基于虚拟机的编程模式,面向对象的应用程序分成多个对象分布到网络中,run-time系统负责代码的放置和自动迁移。此外,Mate<sup>[23]</sup>是建立在TinyOS之上的微型虚拟机,使用类汇编语言写成的程序可以注入到任意节点上,并且能够自我移动。

以上编程模式都是已有的编程模式经过修改,究竟是否能从现有的编程模式得到启发形成新的编程模式还有待研究,但是数据库的编程模型在性能和表达性方面做了折衷,总体优于另外两种模式。传感器网络区别于传统分布式计算的一个主要方面是需要集成存活于自然环境中的组件对象<sup>[31]</sup>。EnviroTrack<sup>[26]</sup>是一个注意到传感器网络和自然环境结合和交互特点的编程模型,该模型主要面向目标跟踪应用。此外,还有一些支持事件驱动和面向大规模嵌入式系统的编程语言,如NesC<sup>[15]</sup>、c@t<sup>[16]</sup>。

## 2.4 动态重编程

传感器网络一般都要配置到不可达区域,并且配置之后会出现新的应用需求,因此传感器网络有必要提供动态可重编程功能,以便增加新的功能或替换原有的系统。由于依靠人工取回节点进行重编程非常繁杂,而且有时不可能重新拿回这些散布出去的节点,因此动态重编程需要依赖于无线代码发布,将新的代码通过无线传输到节点之上。

动态重编程有两个问题需要解决,第一个是代码的可靠传输。由于代码不同于数据,如果代码不完整,就会失去意义而变得不可运行,因此需要研究一种Sink到节点的可靠数据传输<sup>[17]</sup>。另一个是代码在节点上的重建和替换<sup>[18,19]</sup>。

## 3 现有的操作系统

目前已有很多工作围绕传感器网络操作系统展开,并且

出现了一些成型的操作系统。这些操作系统特点各异,本节将对现有的操作系统进行分析,比较其优缺点。

• TinyOS 是由伯克利大学开发的传感器网络专用操作系统,代码量小、耗能少,并且支持并发密集型操作是其主要特点和目标。该系统采用基于组件的体系结构。在这种体系结构下,每个应用程序都是一个组件,一个完整的系统配置是由一个负责调度的调度器和多个组件组成。组件中还可以包含其它有用的组件,而用不到的组件则不会引入到应用系统中,从而达到减少内存需求的目的。此外,TinyOS采用了事件驱动模型,这样可以在很少空间中处理高并发事件,并且能够达到节能的目的,因为CPU不需要主动去寻找感兴趣的事件。其不足是系统没有提供任务之间的同步和通信功能。

• MagnetOS 是由康奈尔大学开发的分布式操作系统,其目标主要是系统能够适应资源的限制以及潜在的网络拓扑变化;减少能源使用,延长网络寿命;提供一个通用和可扩展的操作系统。为此,MagnetOS采用虚拟机的思想,通过运行在边界节点(如sink节点)上的程序分割服务将应用程序分成多个对象分布到网络中,每个对象完成一个特定的功能,并且对每个对象在byte-code级进行重写和插入保持原始程序语义的指令。系统能够根据应用的要求以有利于节能的方式将对象自动放置和迁移到最合适节点上,这样,对象的放置、迁移和对象的执行都不需要用户关心,从而简化了应用程序的开发和减少了通信消耗的能源。这种方式和优点类似于传统分布式系统中的单一系统映像。

• MANTIS 是由科罗拉多大学开发的传感器网络嵌入式操作系统。易于使用和灵活性是其关键目标,以便程序员通过简单的学习就能够快速进行全新的应用开发,而专业研究人员则可以调整和扩展系统来进行高级的研究。为此,MANTIS在操作系统设计和编程上沿用常用的方法,主要表现在采用了经典分层多线程的体系结构和标准C书写的内核和API。与其他操作系统不同的是,MANTIS意识到动态重编程的重要性,提供了无线代码发布功能<sup>[19]</sup>,能够在基站通过无线通信完成节点代码替换,可以更新单个变量、单个线程甚至是整个操作系统。此外,该系统还提供了远程shell供用户登录到sensor节点上察看系统情况。

• SenOS 是一个基于有限状态机模型的操作系统,其目的是为资源受限的传感器节点提供一定的并发性、响应能力和重配置能力。该系统的内核结构是基于有限状态机模型,由三部分组成:(1)由状态序列器(state sequencer)和事件队列组成的内核;(2)状态转换表;(3)回调函数库(call-back library)。状态转换表记录着每一个状态收到一个输入所对应的下一个状态。内核不停地检查事件队列是否有事件到来,如果队列不空,内核从中取出一个事件,根据状态转换表触发一个状态转换,将“机器”从当前状态转换到另一个状态,并调用与状态转换相关的输出函数。这些函数存放在回调函数库中,它和内核一起固化到Flash ROM中,而状态转换表定义了正确的转换和相关的回调函数,是和应用相关的,因此它可以通过系统提供的动态程序装入器装入新的状态转换表。通过这种方式开发应用程序比较简单,但是不够灵活。这个操作系统目前还处于开发中。

• PEEROS<sup>[20]</sup> 是欧洲研究项目EYES的一部分,期望在资源非常受限的环境下开发一个实时的内核,同时能够减轻程序员的负担,并且能够充分利用资源。PEEROS具有可抢占的实时内核,它的调度器是基于EDF(early deadline first)实时调度算法轻量级版本EDF<sup>[21]</sup>调度算法,这样可以在任务的执行得到实时保证的同时,减少资源需求和切换时

间。为了减少内存空间的需求,用户可以将当前用不到的驱动器存放到EEPROM上,当需要时通过模块管理器装入。此外,与其它系统相比,PEEROS在有限的资源条件下提供了实时保证。

#### 4 系统比较

以上所列操作系统特性比较如表1所示。表中所列特性包含了系统体系结构、事件驱动、实时性、编程模式和易用性。其中系统结构是系统的整体框架,影响着其它特性。列出系统是否支持事件驱动模型,以任务能否被抢占来比较实时性。易用性可以分为“好”、“较好”和“最好”三种级别,如果操作系统沿用传统编程模式,用户能够经过很少的学习就能够使用,我们把它列为最好级别,如果系统不是采用传统编程思想,但是提供了良好的编程环境,将其列为较好级别。其它的系统归属于“好”级别。表中所列的编程范例是指程序的逻辑结构。

表1 现有的操作系统特性比较

特性 系统	系统结构	事件驱动模型	实时性支持	编程范例	易用性
TinyOS	组件结构	支持	不可抢占	组件编程	较好
MagnetOS	虚拟机	没提及	没提及	虚拟机	较好
MANTIS	层次结构	支持	不可抢占	传统编程	最好
PEEROS	组件结构	支持	可抢占	传统编程	好
SenOS	状态机	支持	没提及	状态表	好
PALOS*	函数库	支持	不可抢占	传统编程	较好

\* 根据散乱资料整理,参考网站 <http://deerhound.ats.ucla.edu:7771pvs/portal/url/item/6A051BF206024D9CBF01166BAF359CEE>

从表中可以看出,操作系统的体系结构区别很大,一方面因为受应用的影响,另一方面说明系统结构还有待研究。基于事件驱动模型一方面可以获得代码量小,另一方面可以解决有限的可用能源问题,因为采用事件驱动模型可以将代码组织成接到外界事件才响应执行的模块,如果没有事件,系统就可以转入节能模式,因此大部分操作系统都支持事件驱动模型。另外由于采用抢占式实时调度会带来很大的负载,因此大部分操作系统都采用了不可抢占式,PEEROS尝试采用了可抢占式,也说明这方面具有一定的研究空间。

**结论** 无线传感器网络是由很多小型设备自组织而成的用来搜集和传输数据的网络,为我们提供了一种可以随时、随处访问信息和动态参与创建智能环境的工具,它作为人和环境之间的智能中间媒介,已经成为普适计算的关键技术之一。然而,设计一个有效和自适应的传感器网络是一件非常复杂的工程。不同于传统计算平台,无线传感器网络具有独特的体系结构、有限的资源,并且和应用紧密相关,这些特点要求具有一个专用的操作系统,不仅能够运行在轻量级的设备上,具有很少的能源消耗,而且要支持动态的重配置功能,以满足不断改变的环境和应用。

本文从操作系统的角度来分析传感器网络的特点,给出传感器网络操作系统亟需解决的关键技术,综述了传感器网络操作系统的当前研究状况。然而,传感器网络技术还不完善,因此其上的操作系统还有很多问题需要解决,我们认为以下几个方向将会成为传感器网络操作系统未来的研究重点:

(1)研究和开发传感器网络分布式操作系统,为程序员提供单一系统映像,屏蔽底层细节,为用户提供一定的透明度,比如,用户不应当过多地考虑底层的通讯,节点之间的通信应当对用户透明。

(2)研究一个简单有效的编程模式,以满足未来大规模配置网络的需要。

(3)随着微电子技术的发展及传感器网络应用越来越普遍,传感器网络能够提供的服务会越来越多。由于节点的资源有限,因此不可能每个节点上都运行这些服务。如何有效地进行服务描述和服务发现及相关技术会引起人们的注意。

## 参考文献

- 1 Weiser M. The computer for the twenty-first century. Scientific American, Sep. 1991
- 2 Estrin D, Govindan R, Heidemann J, Kumar S. Next century challenges: Scalable coordination in sensor networks MOBICOM, Seattle, 1999. 263~270
- 3 Akyildiz LF, Su WL, Sankarasubramaniam Y, Cayirci E. A survey on sensor networks. IEEE Communications Magazine, 2002, 40(8):102~114
- 4 Abu-Ghazaleh S NB, Heinzelman W. A taxonomy of wireless micro-sensor network models. Mobile Computing and Communications Review, 2002, 1(2):1~8
- 5 Hill J, et al. System Architecture Directions for Networked Sensors. In: Proc. of the Intl. Conf. on Architectural Support for Programming Languages and Operating Systems, 2000
- 6 Kim TH. Designing a State-driven Operating System for Dynamically Reconfigurable Sensor Networks. In: SOC Design Conf., 2003. 40~42
- 7 Abrach E H, et al. MANTIS: System Support For Multimodal Networks of In-situ Sensors. In: 2nd ACM Intl. Workshop on Wireless Sensor Networks and Applications (WSNA), 2003. 50~59
- 8 Barr R, et al. On the Need for System-Level Support for Ad hoc and Sensor Networks. Operating Systems Review, ACM, 2002, 36(2):1~5
- 9 Madden S R, et al. TAG: a Tiny Aggregation Service for Ad-Hoc Sensor Networks. In OSDI 2002, Boston, USA, Dec. 2002
- 10 Santi P, Simon J. Silence is Golden with High Probability: Maintaining a Connected Backbone in Wireless Sensor Networks. In: Proc. IEEE European Workshop on Wireless Sensor Networks, LNCS 2920, Jan. 2004. 106~121
- 11 Sinha Amit, Chandrakasan Ananth. Operating System and Algorithmic Techniques for Energy Scalable Wireless Sensor Networks. In: Proc. of the Second Intl. Conf. on Mobile Data Management, Jan. 2001. 199~209
- 12 Heinzelman W, Chandrakasan A, Balakrishnan H. Energy-Efficient Communication Protocols for Wireless Microsensor Networks. In: Proc. Hawaiian Int'l Conf. on Systems Science, Jan. 2000
- 13 Yao Y, Gehrke JE. The Cougar Approach to In-Network Query Processing in Sensor Networks. Sigmod Record, 2002, 31(3)
- 14 Boulis A, Han C C, Srivastava M B. Design and Implementation of a Framework for Programmable and Efficient Sensor Networks. In MobiSys 2003, San Francisco, USA, May 2003
- 15 Gay D, et al. The nesC Language: A Holistic Approach to Networked Embedded Systems, Intel Research, IRB-2002-019, Nov. 15, 2002
- 16 Seetharamkrishnan D P. c@t: A Language for Programming Massively Distributed Embedded System. [Masters thesis]. Massachusetts Institute of Technology Sep. 2002
- 17 Sankarasubramaniam Y, Akan O, Akyildiz I. ESRT: Event-to-

- Sink Reliable Transport in Wireless Sensor Networks, MobiHoc 03
- 18 Reijers N, Langendoen K. Efficient code distribution in wireless sensor networks. In: Proc. of the 2nd ACM Intl. Conf. on Wireless sensor networks and applications, San Diego, CA, USA, 2003
- 19 Stathopoulos T, Heidemann J, Estrin D. A Remote Code pdate Mechanism for Wireless Sensor Networks; [Technical Report CENS-TR-30]. University of California, Los Angeles, Center for Embedded Networked Computing, Nov. 2003
- 20 Mulder J, Dulman S, et al. PEEROS - System Software for Wireless Sensor Networks. Preprint, Aug. 2003
- 21 Jansen P G, Mullender S J, Havinga P J M, Scholten J. Light-weight EDF scheduling with deadline inheritance; [Technical report (TR-CIT-03-23)]. Centre for Telematics and Information Technology, University of Twente, 2003
- 22 Lifton J, et al. Pushpin computing system overview: A platform for distributed, embedded, ubiquitous sensor networks. In: Intl. Conf. on Pervasive Computing, 2002
- 23 Levis P, Culler D. Maté: A tiny virtual machine for sensor networks. In: Proc. of Intl. Conf. on Architectural Support for Programming Languages and Operating Systems, 2002
- 24 Subramanian L, Katz RH. An Architecture for Building Self-Configurable Systems. In: IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing, Boston, Aug. 2000
- 25 Yu Y, Krishnamachari B, Prasanna VK. Issues in Designing Middleware for Wireless Sensor Networks. IEEE Network Magazine, 2003
- 26 Abdelzaker T, Blum B, et al. EnviroTrack: An environmental programming model for tracking applications in distributed sensor networks, SOSP, 2003
- 27 Vinoski S. Corba: Integrating diverse applications within distributed heterogeneous environments. IEEE Communications Magazine, 1997, 14(2)
- 28 Birrel A, Nelson B. Implementing remote procedure calls. ACM Transactions on Computer Systems, 1984, 2(1)
- 29 Krishnamachari B, Estrin D, Wicker S. Impact of data aggregation in wireless sensor networks. In: Intl. Workshop on Distributed Event-Based Systems, Vienna, Austria, July 2002
- 30 Abdelzaker T, et al. Communication Architecture and Programming Abstractions for Real-Time Embedded Sensor Networks. In: 1st International Workshop on Data Distribution in Real-Time Systems (DDRTS '03), Providence, RI, May 2003
- 31 Addelee M, et al. Implementing a sentient computing system. IEEE Comput., 2001, 34: 50~56
- 32 Tian D, Georganas N D. A Node Scheduling Scheme for Energy Conservation in Large Wireless Sensor Networks. Wireless Communications and Mobile Computing [Journal] May 2003
- 33 Ye F, Zhong G, Lu S, Zhang L. PEAS: A Robust Energy Conserving Protocol for Long-lived Sensor Networks. In ICDCS 2003
- 34 Cerpa R, Estrin D. ASCENT: Adaptive Self-Configuring Sensor Networks Topologies. In: Proc. of the Twenty First Intl. Annual Joint Conf. of the IEEE Computer and Communications Societies (INFOCOM), 2002
- 35 Chen B, Jamieson K, Balakrishnan H, Morris R. Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. In: Proc. of the Sixth Annual Intl. Conf. on Mobile Computing and Networking, 2000
- 36 Heidemann J, et al. Building efficient wireless sensor networks with low-level naming. In: Proc. Symp. Operating Systems Principles, Oct. 2001. 146~159
- 37 Mainwaring A, et al. Wireless Sensor Networks for Habitat Monitoring. In: ACM International Workshop on Wireless Sensor Networks and Applications, Sept. 2002

(上接第6页)

运行环境为:硬件环境:单机、服务器、并行机。操作系统:Windows系列、Linux。数据库:Oracle、Sybase、SQL Server、MySQL等。

结论 针对农村信息化建设中,农村科技信息集成、共享、服务、管理存在的问题,通过对农村科技数据深层次、全方位的需求分析,特别是对共享形式和服务内容的分析,本文提出并实现了农村科技信息共享服务系统。此系统按照统一通信规范格式将广大分散的、异构的信息资源进行记录、分类、整理等定量化、规范化的处理后,集成整合在一起,并有效地加以共享和管理,为用户提供快速、准确、全方位、多层次的农村信息服务。

该系统主要具有下面三个特色,即快速方便的信息集成、充分有效的信息共享以及智能全面的信息服务。

在本项工作的基础上,我们将进一步研制智能教学系统和用户需求自动捕捉和分析等。

## 参考文献

- 1 陈良玉.农村科技信息共享技术与实践.中国农业科学技术出版社,2004
- 2 杜青林.中国农业和农村经济结构战略性调整.北京:中国农业出版社,2003
- 3 彭常.关于信息传输“最后一公里”的研究.中国科协首届学术年会论文集.中国科学技术出版社,1999
- 4 温国泉.浅谈我国现阶段农业信息化服务的发展模式.科技情报开发与经济,2002,12(5):33~34,36
- 5 Uschold M, Gruninger M. Ontologies: Principles, methods and applications. The Knowledge Engineering Review, 1996, 11(2)
- 6 张维明,等.语义信息模型及应用.北京:电子工业出版社,2002
- 7 Cao CG. Extracting and Sharing Medical Knowledge. Journal of computer Science and Tehcnology, 2002, 3
- 8 冯东辉,曹存根.知识界面在NKI中的应用.计算机工程与应用, 2001
- 9 Feng Q, Cao C, Si J, Zheng Y. A Uniform Human Knowledge Interface to the Multi-Domain Knowledge Bases in the National Knowledge Infrastructure. ES2002, 2002. 163~176