Demo Abstract: A Shared Sensor Network Infrastructure

Christos Efstratiou, Ilias Leontiadis, Cecilia Mascolo, Jon Crowcroft
Computer Laboratory
University of Cambridge
Cambridge, UK
{*name.lastname*}@cl.cam.ac.uk

1 Introduction

An increasing number of sensor networks have been deployed to monitor a variety of conditions and situations. At the same time, more and more applications are starting to rely on the data from sensor networks to provide users with (near) real-time information and conditions. This increasing demand of users for accurate information about natural and surrounding phoenomena is creating a business case for application providers.

We argue that while quite ripe networking and duty cycling protocols for sensor networks exist, approaches to sensor network sharing and management are still immature. In particular, typical sensor networks are designed and deployed to serve a single application. Indeed, the common approach in the design of sensor networks is to deploy networks that are *fit-for-purpose* with the primary aim of supporting a single application that belongs to a single authority (usually the owner of the infrastructure)[2]. While this is a sensible approach for short-term and small-scale deployments, in sensor network deployments that consist of thousands of nodes with a life span of multiple years, inducing high costs of deployment and maintenance, the *single-application* approach can lead to inefficient use of resources and low cost-benefit results. Moreover, the requirement for dedicated sensing infrastructure to support new applications belonging to different organisations can lead to unnecessary replication of sensing infrastructure. One example that illustrates this problem is the deployment of cameras on the UK roads and motorways. Typically different authorities (police, highway agency, local city authorities) will deploy their own networks of cameras, occasionally covering the same areas.

In this work we propose a departure from the notion of sensor networks aimed at supporting a single application and serving a single user. We introduce an approach that is based on the decoupling of infrastructure and application ownership. The primary objective of this work is to create a framework that allows sensor network infrastructures to be shared among multiple applications that can potentially belong to different authorities. By achieving this level of decoupling, sensing infrastructures can be viewed as an accessible resource that can be dynamically re-purposed and re-programmed by different authorities, in order to support multiple applications.

A key challenge in realising this vision is the design of a new sensor network architecture that supports multiple applications, dynamically uploaded by different owners and simultaneously running over a shared infrastructure. In this demonstration we illustrate our efforts in exploring this vision. The key characteristics of our approach are:

- A virtualisation layer that is running on each sensor node, abstracts access to sensor resources and allows the management of these resources through policies expressed by the infrastructure owner.
- A runtime environment on each node that allows multiple applications to run inside each node.
- A policy based application deployment that enables multiple application to be deployed over the shared infrastructure.

The following sections offer an overview of our architecture and a description of the demo proposal.

2 System Architecture

The primary aim of this work is to offer an environment that can support multiple applications running on each sensor node. Our objective is to provide an execution environment that *hides* from the running applications the fact that they operate in a shared environment. This allows the developers to build their applications in the conventional way, assuming that they have full access to the resources of the node. This is achieved by maintaining a virtual view of the hardware resources of each node, such as sensors, actuators and network interfaces, for each application.

The hight level architecture of the system is shown in Figure 1. The design of the system is targeted to advanced sensing devices (i.e. imote2, gumstix, smartphones) with enough memory and processing power to support multiple applications. The operating system running on each sensing device is Embedded Linux, a multitasking operating system.

Each application on a sensing device runs inside a sandbox environment where access to hardware resources is only available through the *Virtualisation Runtime*. Previous examples of virtualisation in sensor nodes have focused on fully virtualising the host OS [1]. Instead we allow the applications to run as native process in a controlled environment. The Virtualisation Runtime performs a number of functions:

SenSys'10, November 3–5, 2010, Zurich, Switzerland.
ACM 978-1-4503-0344-6/10/11

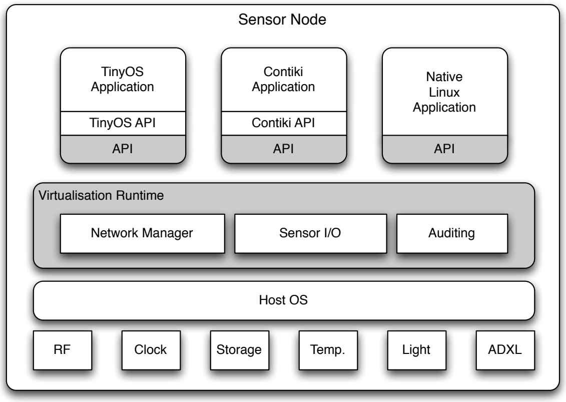

Figure 1. Sensor Node Architecture

Figure 2. Demo topology

- Control access to hardware components. Driven by a set of policies specified by the infrastructure owner, different applications can have different priorities and levels of access to the resources of each sensor node.
- Network traffic encapsulation. Each application running on the network maintains its own communication protocol which is independent of the communication protocol of the host machine. Traffic between instances of a single application is encapsulated inside the traffic of the host machine and isolated from the traffic of other applications.
- Auditing. The runtime records the behaviour of all applications running on each node in terms of resource usage. Such information can be retrieved from the network either for debugging purposes or to inspect for possible irregular behaviour of applications.

Applications can be developed using either conventional sensor network programming environments, like TinyOS and Contiki, or as native Linux processes, *and application developers do not need to be aware of any aspects of the sharing infrastructure but develop in the conventional way*. Support for the latter is achieved through a well defined API that allows applications to access hardware resources through the virtualisation runtime. In order to support TinyOS and Contiki applications we implemented a set of virtual drivers/components that were incorporated in the two SDKs. The virtual drivers use our API to access the hardware through the runtime.

3 Demonstration Proposal

In this demonstration, conference delegates will be able to observe the sharing of a sensor network infrastructure by multiple sensing applications. The demonstration is built around the scenario of a sensing infrastructure installed in an office building. Ten imote2 sensor nodes will be deployed on top of a building plan to illustrate the physical topology of the sensor network.

During the demonstration three separate applications will be deployed concurrently on the network:

- *Environment Monitoring*: This application collects environmental readings (temperature, humidity) and reports these readings to the sink. The application will be deployed on all nodes of the network. A back-end application will show in a graphical format the data that is collected.
- *Occupancy Monitoring*: This application utilises the ambient light sensor to infer the occupancy of rooms. The application will be deployed only on sensors that are located in office spaces. A back-end application will show occupancy changes on the building plan.
- *Fire Evacuation*: This application utilises the temperature sensor to monitor the progress of a fire in the building. The application will be deployed on demand illustrating a special case of emergency. The underlying infrastructure will give priority to this application's traffic over the previous applications.

The layout of the demonstration is shown in Figure 2. The sensor node neighbourhood is pre-configured to represent a realistic multi-hop topology. A laptop acts as a sink for the sensor network and it hosts the back-end portion of the deployed applications. A second laptop illustrates in real-time network events, such as applications deployed on each node and network traffic, utilising the Auditing component of the infrastructure.

Through this demonstration the delegates will be able to observe how multiple applications can run on a shared infrastructure. They will see examples of selective deployment of applications on parts of the infrastructure and the effect of priorities and policies on the behaviour of the applications.

4 References

[1] P. Levis and D. Culler. Mate: A tiny virtual machine for sensor networks. In *International Conference on Architectural Support for Programming Languages and Operating Systems, San Jose, CA, USA*, Oct 2002.

[2] J. Yick, B. Mukherjee, and D. Ghosal. Wireless sensor network survey. *Computer Networks*, 52(12):2292–2230, August 2008.