

# Deployment Support for Wireless Sensor Networks

Matthias Dyer, Jan Beutel, and Lennart Meier

Computer Engineering and Networks Laboratory, ETH Zurich, Switzerland  
{`dyer,beutel,meier`}@tik.ee.ethz.ch

**Abstract.** We present the concept of Deployment-Support Networks (DSN), a tool for the coordinated deployment of distributed sensor networks. A DSN supports the development, debugging, monitoring, and testing of sensor-network algorithms and applications. With our implementation of the DSN on the BTnode rev3 platform, we have proven that the concept scales well to a large number of nodes.

## 1 Introduction

The recent focus on wireless sensor networks (WSNs) has brought about many different platforms such as the BTnodes [1] or the Berkeley Motes [2]. Researchers are successfully using these platforms in the development of a multitude of sensor-network applications and in the deployment of demonstrators. However, setups with more than 10–20 nodes have shown to be hard to manage, especially when situated in a realistic physical environment.

The key problem is the lack of appropriate support tools for large numbers of distributed devices. Coordinated methods for testing, monitoring, programming, and debugging of WSN applications in realistic scenarios are missing so far. Existing methods commonly used for developing and testing embedded systems are not sufficient for WSNs since they do not meet the new requirements.

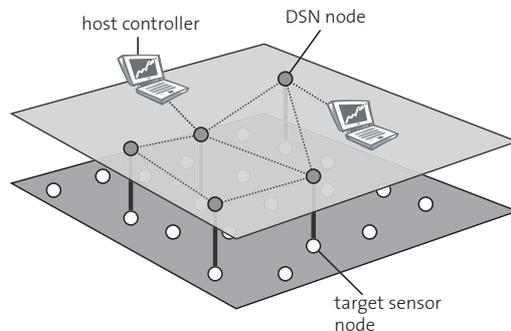
A step in the right direction are the simulators [3] and testbeds [4] that have been built for WSNs. Simulators abstract the WSN hardware. They incorporate a more or less accurate model of the computation and communication and are very useful in the early phase of development where basic concepts of collaborative applications and protocols need to be tested. However, for more complex and advanced development, the models used in simulators are often too simplistic [5]. For this reason, researchers have built testbeds with real devices. Existing testbeds consist of a collection of sensor nodes that are connected to a fixed infrastructure, such as serial cables or ethernet boxes. Testbeds are more realistic than simulators because they use the real devices and communication channels. The problem that remains is that the conditions in the field where the WSN should be deployed in the end can be significantly different from the testbed in a laboratory. In particular, with a cable-based infrastructure it is impossible to test the application with a large number of nodes out in the field.

To address these issues, we present the deployment-support network (DSN), a tool for developing, testing, programming, and monitoring WSNs in a realistic

environment. The term *deployment support*, as it is used in this paper, does not only stand for supporting the act of deploying nodes in the field, but for the whole process of development and testing in the field.

## 2 Deployment Support Network

The main idea of the DSN is to replace the cable based infrastructure of the testbeds with a wireless infrastructure. This is achieved by attaching the sensor node to a wireless embedded node, the DSN-node. The DSN-nodes form a self-maintained network that provides deployment support services. This network is referred to as *Deployment-Support Network* (see Fig. 1). The DSN does not replace the WSN that is used for the application. DSN and WSN coexist during the development phase. When this phase is completed, the DSN nodes are not needed anymore and are detached from the sensor nodes.



**Fig. 1.** Abstract view of a DSN



**Fig. 2.** The BTnode rev.3 platform.

Conceptually, there are many services that can be offered by a DSN. However, we concentrate here on four services that we believe are the most important.

**Wireless Node Access:** As a general service, the DSN provides access to the sensor nodes with a wireless backbone network. One or more host computer are used to connect to the network and to communicate with the target sensor nodes. This service includes algorithms for the construction and maintenance of a network, the networking services and protocols such as broadcast and multi-hop transport and higher-level services such as finding nodes and obtaining information on the topology.

**Serial Tunnel:** The DSN replaces the serial cables. It offers a serial connection from a host computer to a target sensor node. More than one connection can be used simultaneously, but the maximal throughput of the backbone network limits the traffic on shared links. The multi-purpose serial connections

are widely accepted as standard I/O for sensor nodes. Having the serial connection tunneled over a wireless network allows the host controller to communicate with the sensor nodes as over a serial cable. Possible applications that use this service are, e.g., remote debugging, interactive terminal sessions, or any application-specific serial data transport.

**Target (Re)Programming:** Remote programming is important since the software running on the target sensor nodes is still under development and many test cycles with small updates are needed. With this service, the sensor nodes can remain deployed while being reprogrammed.

**Event-based Monitoring:** While the serial tunnel is a connection-oriented service, event-based monitoring is connectionless. Triggered by the sensor nodes, the DSN nodes send events to a host controller.

Alternatively, the described services could be implemented on the sensor nodes themselves, and instead of using the DSN, the existing wireless network could be used to transport debug, monitoring, and control information. The benefit of the DSN in over this alternative or a cable-based solution are obvious:

- The DSN and the WSN have different requirements on the wireless interface. WSNs are optimized for energy efficiency. The WSN has to meet the bandwidth requirement of the application which is typically reported as being in the order of 10 kbps. A DSN node can afford a more powerful wireless interface that is optimized for deployment support.
- The WSN cannot be used to reliably transport debug and control information when the application and protocols on the sensor node are still under development.
- Using the WSN for remote programming is dangerous as nodes can become unreachable if the programming fails or the new program version is erroneous. Because the DSN nodes are not reprogrammed, they are always reachable and can reprogram the sensor node at any time.
- The DSN nodes are small and battery-operated. Therefore they can easily be deployed almost everywhere. The logistical overhead of deploying a WSN with a DSN is minimal.
- The DSN solution scales well to a large number of nodes. The DSN bandwidth limits the number of concurrently active connections and not the number of nodes in the DSN in general.
- The DSN services unburden the sensor nodes. Less debugging and control code has to be implemented on a sensor node if it is attached to a DSN node.

### 3 Implementation and Results

We have implemented a prototype of a DSN on the BTnode rev3 platform (see Fig. 2). We use Bluetooth as the wireless interface for the DSN. A distributed topology-control algorithm has been implemented that forms and maintains automatically a tree-based topology with all reachable DSN nodes. The networking

services are implemented in a multi-hop middleware layer. On top of this middleware are the services that handle the serial tunnels and monitor the target sensor node. Our BTnut system software consists of a port of the embedded multi-threaded OS “Ethernut” and a Bluetooth stack.

The host computer runs a monitoring and control tool with which the engineer can obtain information on the topology, control the nodes, or open and close serial tunnels.

In a case study [6] we have measured parameters of our prototype such as the delay per hop on serial tunnels and the initial-topology-construction speed. The measurements are performed on automatically formed networks with up to 71 BTnodes. To our knowledge, these are the largest connected Bluetooth networks that have been reported so far.

## 4 Conclusion

With the concept of a deployment-support network, we have introduced a valuable tool for the coordinated deployment of distributed sensor networks. In contrast to existing test setups in laboratories, the DSN approach allows large-scale deployment without losing the ability to observe and control all nodes and without the burden of fixed, wired infrastructure or changes to the target system. Scalable communication to large populations of WSN target devices embedded in their designated physical environment will allow to investigate the performance of live sensor systems to enable detailed comparisons and validation.

## References

1. Beutel, J., Kasten, O., Mattern, F., Römer, K., Siegemund, F., Thiele, L.: Prototyping wireless sensor network applications with BTnodes. In: Proc. 1st European Workshop on Sensor Networks (EWSN 2004). Volume 2920 of Lecture Notes in Computer Science., Springer, Berlin (2004) 323–338
2. Hill, J., Culler, D.: Mica: A wireless platform for deeply embedded networks. *IEEE Micro* **22** (2002) 12–24
3. Levis, P., Lee, N., Welsh, M., Culler, D.: TOSSIM: Accurate and scalable simulation of entire TinyOS applications. In: Proc. of the 1st int’l conference on Embedded networked sensor systems (SenSys), ACM Press, New York (2003) 126–137
4. Girod, L., Elson, J., Cerpa, A., Stathopoulos, T., Ramanathan, N., Estrin, D.: EmStar: A software environment for developing and deploying wireless sensor networks. In: Proc. USENIX 2004 Annual Tech. Conf. (2004) 283–296
5. Kotz, D., Newport, C., Gray, R., Liu, J., Yuan, Y., Elliott, C.: Experimental evaluation of wireless simulation assumptions. In: Int’l Workshop Modeling Analysis and Simulation of Wireless and Mobile Systems (MSWiM), ACM Press, New York (2004) to appear
6. Beutel, J., Dyer, M., Meier, L., Thiele, L.: Scalable topology control for deployment-support networks. In: The Fourth International Conference on Information Processing in Sensor Networks (IPSN). (2005) to appear