Millennial Net

Commercial- and Industrial-Class Wireless Sensor Networks

MeshScape™ Technical Overview

A Millennial Net White Paper

Introduction

MeshScape™ is a complete wireless sensor networking system that delivers OEMs and system integrators all of the software, hardware, and tools to quickly and cost-effectively develop and deploy embedded wireless sensor networking applications. The MeshScape system includes :

— **MeshScape Network Protocol:** The ultra-efficient, highly scalable, self-organizing networking software is based on Persistent Dynamic Routing™ technology and supports a variety of standard network topologies .

— **Data models:** The MeshScape system provides built-in support for data movement profiles to speed development. These data models optimize the network for an application's specific data requirements and support a variety of classes for collection and bi-directional dialogue data models.

— **APIs:** Application APIs streamline development providing input/output functions for sensor and application integration.

Persistent Dynamic Routing™

Millennial Net has developed and optimized its protocol to address the unique characteristics and challenges associated with wireless sensor networking. The end result is a networking system and associated protocol that is highly scalable, ultra-efficient, and extremely responsive and resilient in dynamic environments. The Millennial Net protocol for wireless sensor networks that provides the industry's longest battery life at sensor nodes while delivering data over fault-tolerant links with end-to-end redundancy. The Millennial Net protocol is based on a set of techniques including, Persistent Dynamic Routing for reliable and scalable wireless sensor networks. When forming an ad hoc sensor network, Persistent Dynamic Routing requires minimal overhead for requesting and establishing connectivity without relying on the bandwidth-consuming flooding technique.

How it Works

Persistent Dynamic Routing provides a mechanism for the network to ensure reliable data transmission without dropping data packets. Combined with the technique of dynamic route discovery that discovers the best route for packet delivery on the fly, Persistent Dynamic Routing enables a level of scalability and power efficiency that other networking systems cannot achieve.

Almost every existing ad hoc network protocol assumes some level of static status of the network. For example, the route discovery process of AODV assumes there is at least a short duration during which a "snapshot" of the complete route to the destination is possible. The data packet of DSR carries the full route information in itself, which assumes the existence of a "full" route at that moment.

In the case of relatively static network with low fluctuation and interference, this assumption can hold with reasonable level of success. But in a highly dynamic environment, the assumption of this kind of "quasi-static" status does not hold. In other words, the network may be continuously changing so that it is impossible to establish a full route from the source to the destination at a point in time. In this case, traditional routing algorithms such as AODV or DSR can present difficulties. For example, in an AODV system, the source will keep sending the route discovery packet but will not get a definite route response from the destination, which will result in continuous flooding of network with route discovery packets. As a result, the data packet will not be even sent into the network since route discovery process is incomplete. More route discovery packets translate into more overhead. This problem will be even more serious in a large-sized network; since the route discovery is essentially a flooding process, the efficiency of the network will drop significantly.

The gateway establishes tree structure for dynamic addressing. Route discovery packet broadcasts through the network establishing mesh routing (AODV) which floods the network. Route response packet broadcasts to validate the route. Data packet sent to the gateway.

With Persistent Dynamic Routing, the data packet does not need to wait until the route discovery process grabs a "full" route at a moment in time. A data packet is released and navigates through the network with the best knowledge it has collected from its neighbors at that moment. It works in a manner similar to the mechanism of navigating a maze without any prior knowledge of the maze. The data packet does not wait until the full route is confirmed; rather, it starts navigating the network with whatever information it has about the destination.

Persistent Dynamic Routing can significantly decrease the overhead of packet delivery in a highly dynamic network since it does not send excessive numbers of route discovery packets nor does it use proactive route updates. Also, the route discovery packet in Persistent Dynamic Routing does not go more than one hop in each discovery process, resulting in less flooding of the network. In practice, flooding is used only once at the very beginning of the network formation and, from then on, route discovery is only done in the local area to collect knowledge on the best route to the destination. This "best knowledge" has no guarantee that it is correct, and the data packet does not "ask" for that kind of guarantee. In this sense, Persistent Dynamic Routing is based on the probabilistic rather than deterministic approach. In a relatively static network, the higher probability that the destination matches the actual deterministic route would give Persistent Dynamic Routing the same level of performance as AODV, if not better. In highly dynamic environment, Persistent Dynamic Routing produces significantly less overhead in packet delivery than the AODV flooding approach.

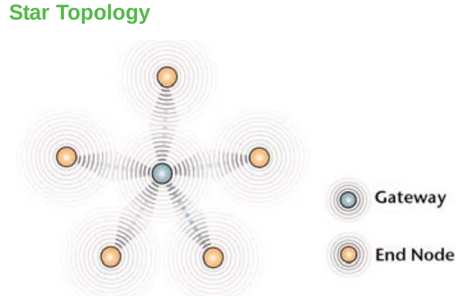
Topologies

The MeshScape system supports a choice of standard topologies including star, mesh, and star-mesh hybrid. The selection of topology—configuration of the hardware components and how the data is transmitted through that configuration—is driven by the requirements of the application.

Star

The star topology is a single-hop system in which all wireless sensor nodes are within direct communication range—usually 30-100 meters—to the gateway. In this configuration all sensor nodes are end nodes and the gateway serves to communicate data and commands to the end nodes. The gateway is also used to transmit data to a higher-level control or monitoring system. The end nodes do not pass data or commands to each other; they use the gateway as a coordination point.

Star Topology

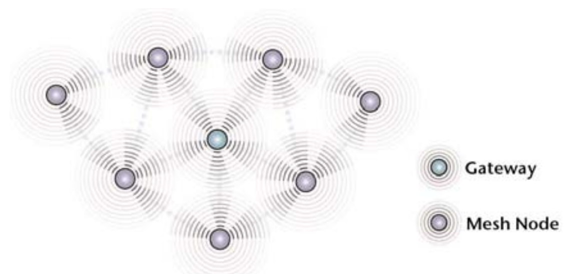


The star topology delivers the lowest overall power consumptions but is limited by the transmission distance of the radio in each endpoint back to the gateway, typically 30 to 100 meters in the ISM band. There are also no alternate communication paths between the endpoints; should a path become obstructed, communication with the associated endpoint will be lost.

A medical monitoring PAN (personal area network) networks medical devices measuring patient information such as blood pressure and glucose level wirelessly transmit the data to a PDA by a doctor performing rounds. The star topology is well-suited to this application. The range for this application is short and the network is fairly small, consisting typically of an average of five devices.

Mesh

Mesh Topology



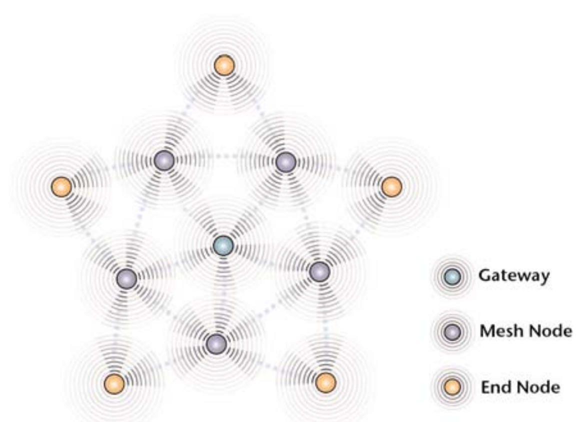
Mesh topologies are multi-hopping systems in which all wireless sensor nodes are mesh nodes and communicate with each other to hop data to and from the sensors and the gateway. Unlike the star architecture where the nodes can only talk to the gateway, the nodes in a mesh topology can also hop messages among other mesh nodes.

The propagation of sensor data through the mesh allows a sensor network to be extended, in theory to an unlimited range. The mesh network is also highly fault tolerant since each sensor node has multiple paths back to the gateway and to other nodes. If a sensor node fails, the network will reconfigure itself around the failed node automatically.

A mesh topology is appropriate for an energy management application that transmits temperature data to a controller to monitor energy usage. The sensors are integrated with router nodes rather than endpoints to form a complete mesh so that each node can both capture sensor data conduct routing. Power is not a critical issue in this application, nor is latency, but the ability to expand the distance of the network in large buildings is extremely important.

Star-Mesh Hybrid

Star-Mesh Hybrid Topology



The star-mesh hybrid takes advantage of the low power and simplicity of the star topology, as well as the extended range and self-healing nature of a mesh topology. The star-mesh hybrid organizes end nodes in a star topology around mesh nodes which, in turn, organize themselves in a mesh network. The mesh nodes serve both to extend the range of the network and to provide fault tolerance. Since end nodes can communicate with multiple mesh nodes, the network reconfigures itself around the remaining mesh nodes if one fails or if a radio link experiences interference.

The star-mesh network offers the highest degree of sensor node mobility and flexibility for rapid changes to the network population and the lowest overall power consumption for networks that need to stretch beyond 30-100 meters. For these reasons, the star-mesh hybrid is proving to be a logical choice for many implementations of wireless sensor networking.

An industrial gas pad application that uses sensors in tanks to monitor fill levels requires a star-mesh hybrid architecture. A star network is deployed at each tank to capture the sensor data which is then routed through a mesh to extend the network's range across the large fields to the gateway.

Data Models

The data model describes the way in which the data flows through the network. The data model differs from topology. Topology refers to the configuration of the hardware components and how the data is transmitted through that configuration. The data model, on the other hand, is a function of the application and describes the flow of the data in terms of how that data is used. There are two broad categories of data models: data collection models and bi-directional dialogue models. The MeshScape system provides built-in support for data movement profiles to speed development. These data models optimize the network for an application's specific data requirements and support a variety of classes for collection and bi-directional dialogue data models.

Data collection models describe monitoring applications where the data flows primarily from the sensor node to the gateway. There are three common data collection models: periodic sampling, event driven, and store and forward. Bi-directional dialogue application classes are characterized by a need for two-way communication between the sensor/actuator nodes and the gateway/application. There are two common data collection models: polling and on-demand.

Periodic Sampling

For applications where certain conditions or processes need to be monitored constantly, such as the temperature in a conditioned space or pressure in a process pipeline, sensor data is acquired from a number of remote sensor nodes and forwarded to the gateway or data collection center on a periodic basis. The sampling period mainly depends on how fast the condition or process varies and what intrinsic characteristics need to be captured.

In many cases, the dynamics of the condition or process to be monitored can slow down or speed up from time to time. Therefore, if the sensor node can adapt its sampling rates to the changing dynamics of the condition or process, over-sampling can be minimized and power efficiency of the overall network system can be further improved.

Another critical design issue associated with periodic sampling applications is the phase relation among multiple sensor nodes. If two sensor nodes operate with identical or similar sampling rates, collisions between packets from the two nodes is likely to happen repeatedly. It is essential that sensor nodes can detect this repeated collision and introduce a phase shift between the two transmission sequences in order to avoid further collisions.

Event Driven

There are many cases that require monitoring one or more crucial variables immediately following a specific event or condition. Common examples include fire alarms, door and window sensors, or instruments that are user activated. To support event-driven operations with adequate power efficiency and speed of response, the sensor node must be designed such that its power consumption is minimal in the absence of any triggering event, and the wake-up time is relatively short when the specific event or condition occurs. Many applications require a combination of event driven data collection and periodic sampling.

Store and Forward

In many applications, data can be captured and stored or even processed by a sensor node before it is transmitted to the gateway or base station. Instead of immediately transmitting every data unit as it is acquired, aggregating and processing data by remote sensor nodes can potentially improve overall network performance in both power consumption and bandwidth efficiency. One example of a store-and-forward application is cold-chain management where the temperature in a freight container is captured and stored; when the shipment is received, the temperature readings from the trip are downloaded and viewed to ensure that the temperature and humidity stayed within the desired range.

Polling

Controller-based applications use a polling data model. In this model, there is an initial device discovery process that associates a device ID with each physical device in the network. The controller then polls each device on the network successively, typically by sending a serial query message and waiting for a response to that message. For example, an energy management application would use a polling data model to enable the application controllers to poll thermostats, variable air volume (VAV) sensors, and other devices for temperature and other readings.

On-Demand

The on-demand data model supports highly mobile nodes in the network where a gateway device enters the network, automatically binds to that network and gathers data, then leaves the network. With this model, one mobile gateway can bind to multiple networks and multiple mobile gateways can bind to a given network. An example of an application using the on-demand data model is a medical monitoring application where patients in a hospital wear sensors to monitor vital signs and doctors access that data via a PDA that is a mobile gateway. A doctor enters a room and the mobile PDA automatically binds with the network associated with that patient and downloads vital sensor data. When the doctor enters a second patient's room, the PDA automatically binds with that network and downloads the second patient's data.

APIs

The MeshScape system delivers a fully documented set of APIs to streamline the development process by providing input/output functions for sensor and application integration, including:

— **Core functions:** Session management, enumeration of network drives, static and dynamic device attributes, uUniversally supported device properties, event notification, and immediate delivery of packet from monitor to destination device

— **Standard I/O peripherals:** Static device attributes, analog-to-digital conversion, digital input/output, serial data interface (UART), and ADC raw data

— **Supplementary functions:** obtain text description for error codes and convert between ID structures and text representation

About Millennial Net

Millennial Net develops commercial- and industrial-grade wireless sensor networking systems that enable OEMs and systems integrators to quickly and cost-effectively implement wireless sensor networks. These networks enable the remote monitoring and management of critical devices while providing data to enable more informed decision-making, better control and increased revenue opportunities. Millennial Net's patented ad hoc, self-organizing wireless sensor networking MeshScape™ system leads the industry in power efficiency, support for dynamic systems and mobile sensors, reliability, and scalability. Millennial Net also leads the industry in real-world deployments with networks installed across commercial building and industrial environments. Millennial Net is headquartered in Chelmsford, MA.

For more information, visit www.millennialnet.com.

Millennial Net

285 Billerica Road
Chelmsford, MA 01824
Tel.: (978) 569 1921
info@millennialnet.com
www.millennialnet.com