Over the past decade, Meshdynamics has supplied customized versions of our mesh networking software to OEMs: numerous military contractors, industrial mining equipment manufacturers and industrial network equipment manufacturers. **Customers**

Some representative applications shown below (Courtesy Sharp Electronics, Japan)

The core mesh software was ported to run on boards (processors) and radios of the customer's choosing. Meshdynamics system integrators, working directly with the OEM licensee, developed the finished mesh-enabled product. The intellectual property and trade secrets related to their products was thus preserved as a separate software tree.

Post 2014, Meshdynamics developed and field tested MeshSuite™ modules to accelerate time to market for OEMs requiring mesh network connectivity to be baked into their connected product offerings. Modules include simulation and test automation tools specific to mesh networking and source code for exemplary board-radio ensembles.

Patented technology, licensed with MeshSuite includes:

- Dual Radio meshed backhaul with **proven scalable performance**.
- User level publish-subscribe messaging for apps. See Confluence pages for more Logical Radios—dynamically assign logical radio modes to **multi-radio** or **single -radio**
- Dynamic configuration of “**Model number**” based on changing environment
- Disruption Tolerant – e.g. proactive fail over, no packet loss. **DTN Hospital**
- Test bed in a box -- with real and virtual mesh nodes and “app” framework.
- Access to experienced system integrators – accelerated time to market.

**Collaborations Welcomed**  **Dynamic Tree Topologies**  **Self Classified Chirps**  **Smart Simulations**
The architecture includes Mesh Networking algorithms (meshap) and an application framework.

The Mesh Networking architecture defines a design approach where the proprietary Mesh Networking algorithm will be fully integrated with the Linux based mac80211 architecture.

It provides the complete abstraction of the proprietary Mesh Networking algorithm (meshap) from the underlying device drivers. If the underlying device driver changes, the meshap continues to provide services without any impact and no modification is required in the code.

On the top of meshap there is application layer in user space along with its management layer.

The NMS API uses the application layer management plane for J2EE work flow integration.
On the top of meshap networking the application layer resides in user space along with its management layer. Applications – on the node or in the cloud - can subscribe to different routing strategies or devise their own messaging profiles.

The NMS API uses the application layer management plane for J2EE work flow integration.
Python based RIDE Robot test framework used for both unit level and system level tests
Details under MeshSuite Confluence pages.
An evolutionary wireless network concept has been prototyped for military customers, where the simulation world and real world are in lock step. Machine learning begins at the simulation end, and drives auto tuning algorithms to meet specific difficult use case scenarios.

The refined scripts and applications are then uploaded to real mesh nodes to complete the iterative loop. The mesh nodes, through rudimentary publish/discover/subscribe messaging, emulate SDN philosophy for low power devices at the edge.

Nodes resident trusted applications publish real time sensor streams for consumption primarily by other apps in the network. All streams are aggregated at the root nodes and transmitted to big data servers for analysis -- when connected.

Applications customize/tune network behavior, communicating with supervisory apps running on the cloud or a local computer. The distribution of resources (power, bandwidth) is managed collaboratively through the application layer. This keeps the Kernel simple.

Java source code for the NMS and Heart Beat protocol are offered as an illustration of how applications on mesh nodes may publish and listen with their own proprietary heart beat protocols.
**MeshDynamics Licensed Intellectual Property**

J01. Issued 08.07.2017 Terse Message Networks
J02. Issued 06.07.2016 Chirp Networks
J03. Issued 02.09.2016 Chirp Networks
J04. Issued 10.27.2015 Collaborative Logistics Ecosystem: Extensive Framework For..
J05. Issued 06.02.2015 Real Time Packet Transforms To Avoid Re-Transmissions
J06. Issued 04.28.2015 Self Forming VoIP Network
J07. Issued 03.10.2015 Persistent Mesh for Isolated, Mobile and Temporal Networking
J08. Issued 12.30.2014 Chirp Networks
J09. Issued 08.27.2013 Persistent Mesh for Isolated, Mobile and Temporal Networking
J10. Issued 08.20.2013 Real Time Packet Transforms To Avoid Re-Transmissions
J11. Issued 07.02.2013 Self Forming VoIP Network
J12. Issued 06.11.2013 High Performance Mesh Networks- Switch Stack Paradigm
J13. Issued 02.22.2011 Mobility Extensions for Wireless Multiple Radio Mesh
J15. Issued 09.01.2009 Managing Jitter and Latency in Wireless LANs
J16. Issued 09.02.2008 High Performance Wireless Networks Using Distributed Control

J17. Filed 02.28.2018 Evolutionary Wireless Networks
J18. Filed 02.15.2018 Chirp Networks (Continuation)

*What is covered in these patents and part of Mesh Suite Licenses*

**Dynamic Tree Topology**
- Many Hops, Mobility
- Disruption Tolerance
- Learning Resilience

**Abstractions**
- Radio Card Agnostic
- Single & Multi-radio
- Application Pub-Sub

**Real Time “Fog” Computing**
- Real Time Data Streams
- Time Sensitive Networks
- Collaborative Scheduling
- PLC replacement in Node
- Industrial Internet Integration