Small, Dumb, Cheap, and Copious – the Future of the Internet of Things

by Francis daCosta, Founder and CTO, MeshDynamics, Inc.

Abstract

Over the next decade, billions of interconnected devices will be monitoring and responding to transportation systems, factories, farms, forests, utilities, soil and weather conditions, oceans, and other resources.

The unique characteristic that the majority of these otherwise incredibly diverse Internet of Things (IOT) devices will share is that they will be too *small*, too *dumb*, too *cheap*, and too *copious* to use traditional networking protocols such as IPv6.

For the same reasons, this tidal wave of IOT devices cannot be controlled by existing operational techniques and tools. Instead, lessons from Nature's massive scale will guide a new architecture for the IOT.

Taking cues from Nature, and in collaboration with our OEM licensees, MeshDynamics is extending concepts outlined in the book "*Rethinking the Internet of Things*" to real-world problems of supporting "smart: secure and scalable" IOT Machine-to-Machine (M2M) communities at the edge.

Simple devices, speaking simply

Today companies view the IOT as an extension of current networking protocols and practices. But those on the front lines of the Industrial Internet of Things are seeing problems already:

> *"While much of the ink spilled today is about evolutionary improvements using modern IT technologies to address traditional operational technology concerns, the real business impact will be to expand our horizon of addressable concerns. Traditional operational technology has focused on process correctness and safety; traditional IT has focused on time to market and, as a recent concern, security. Both disciplines have developed in a world of relative scarcity, with perhaps hundreds of devices interconnected to perform specific tasks. The future, however, points toward billions of devices and tasks that change by the millisecond under autonomous control, and are so distributed they cannot be tracked by any individual. Our existing processes for ensuring safety, security and management break down when faced with such scale. Stimulating the redevelopment of our technologies for this new world is a focal point for the Industrial Internet Consortium."*

Industrial Internet Consortium *Quarterly Report February 2016*

A truly *scalable* IOT architecture mandates a different worldview: one where the machines take care of themselves and only involve humans for exceptions. Simple devices, speaking simply.

This world of machine-to-machine interaction will be much more like birdsong or the interactions of social insects such as bees and ants than it will be like TCP/IP and WiFi.

At the edge of the network are simple Devices. These edge devices simply "chirp" their bits of data or listen for chirps directed toward them. The *vast numerical majority of devices* will simply speak and listen in tiny bits of data ("small" data). In an IOT universe, these small and seemingly unsecure (covered later) chirps will propagate upstream and up the food chain to cloud-based integration points, a.k.a Big Data subscribers, see Figure 1.

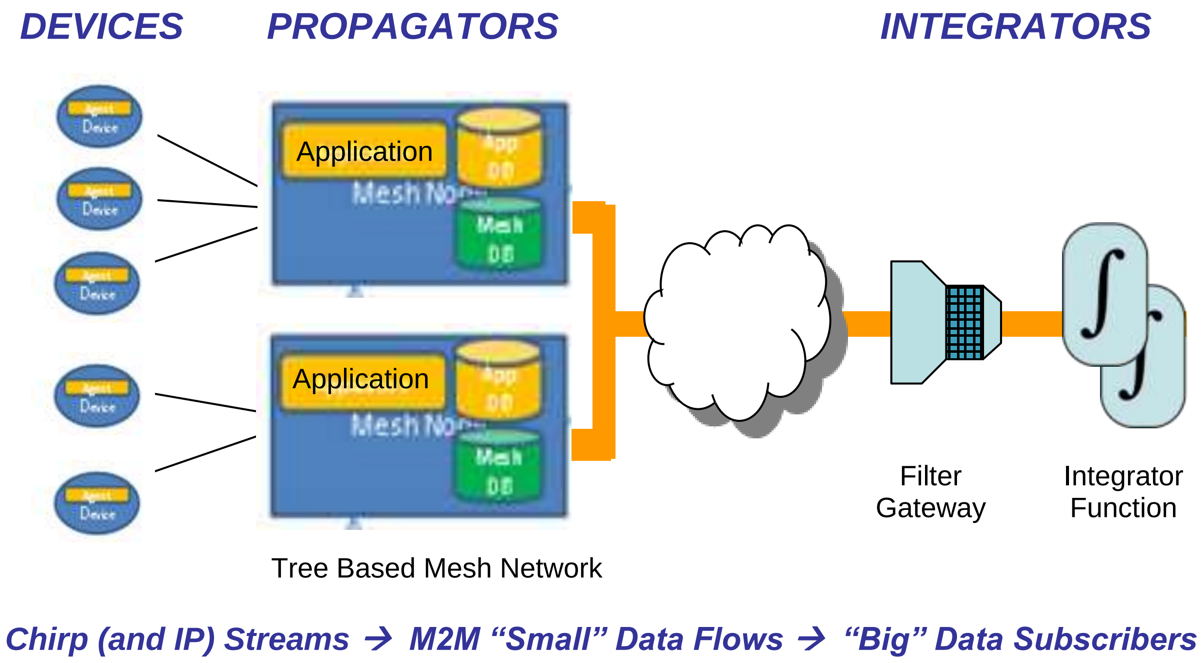


FIGURE 1: Scalable and Secure Basic Architecture for IOT

Like Nature treats pollen, the (scalable) IOT must treat any single chirp as truly "best effort" – so heavy broadcast storms caused by an external event will die out pretty quickly. IOT chirps are digital pollen – lightweight, broadly "published", with meaning only to "interested" receivers/subscribers. Mimicking Nature, the IOT is receiver-centric, not sender-centric (e.g. IP).

Seasonal or episodic broadcast storms from billions of these end devices are much less of a problem because chirps are small and individually uncritical.

Pollen is lightweight because it is receiver-oriented. Security is inherent in its "packet" structure. Also, no individual chirp message is critical so there's no need for error-recovery or integrity-checking overhead (except for basic checksums).

Each IOT chirp message simply has some short and simple markers, a short and variable data field, and a checksum. As described in my book, the simplest chirps may be only 5 bytes (contrast this with 40 bytes for the smallest sender-oriented IPv6 packet). [Slides]

In contrast to the traditional Internet, error-checking, routing, higher-level addressing, or anything of the sort *are not needed*. Edge devices are fairly mindless "worker bees" existing on a minimum of data flow. This will suffice for the *overwhelming majority* of devices connected to the IOT. (And for those more-sophisticated applications where higher-level protocols are still needed and justified by human interaction, IPv6 will do nicely.)

Chirps are what IP Datagrams were meant to be. The bandwidth savings are immediately obvious, but pale in comparison to the reduction in memory, processing, and power consumption compared to running an IP stack at each of myriad end devices. The cost and complexity burden on the end devices will be very low, as it must be in the IOT.

The basic concept of chirps is not new. Terse M2M messaging is prevalent in all of our purpose-built end devices and products that communicate – your TV remote, your car subsystems, networked factories etc. Terse M2M messaging is how machines have communicated since 8-bit microcontroller days. Challenges lies in scaling securely what already works, but not reinventing it.

Publish, subscribe, and discovery for the edge

So if simple devices aren't capable of protocol intelligence, it must reside somewhere. The major elements of that somewhere are the Level II **Propagator** nodes

(Figure 1 and 2). These are like familiar networking equipment such as routers and access points, but they operate in a different way.

Propagators listen for data "chirping" from any device. Based on a simple set of “markers” in the chirps (described below), propagator nodes decide how to broadcast these chirps to other propagator nodes and on to the higher-level **Integrator** subscribers.

In order to scale to the immense size of the Internet of Things, these propagator nodes must be capable of a great deal of *discovery and self-organization*. They must recognize other propagator nodes within range, set up simple routing tables of adjacencies, and discover likely paths to the appropriate integrators.

The key is building a logical tree-like topology from physically meshed propagators. The topology algorithms have been tested and described in (more geeky) MeshDynamics patents. [More]

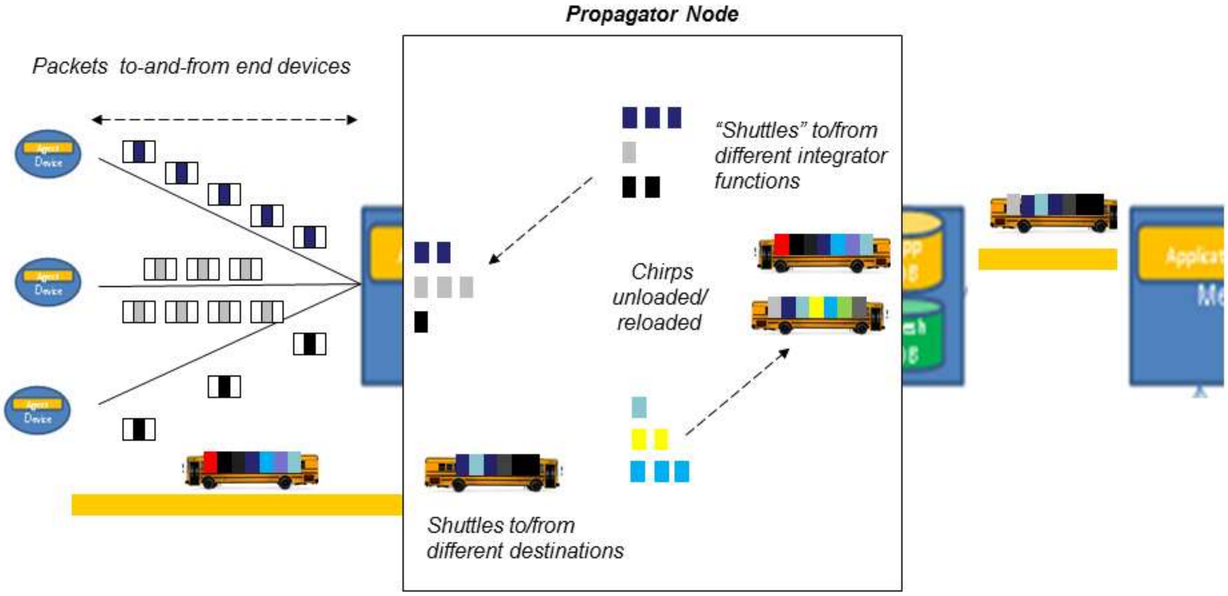

Propagator Node
Packets to-and-from end devices
Agent Device
Agent Device
Agent Device
“Shuttles” to/from different integrator functions
Chirps unloaded/ reloaded
Shuttles to/from different destinations

FIGURE 2: Applications for Aggregation, Pruning and Real Time M2M “Shuttles”

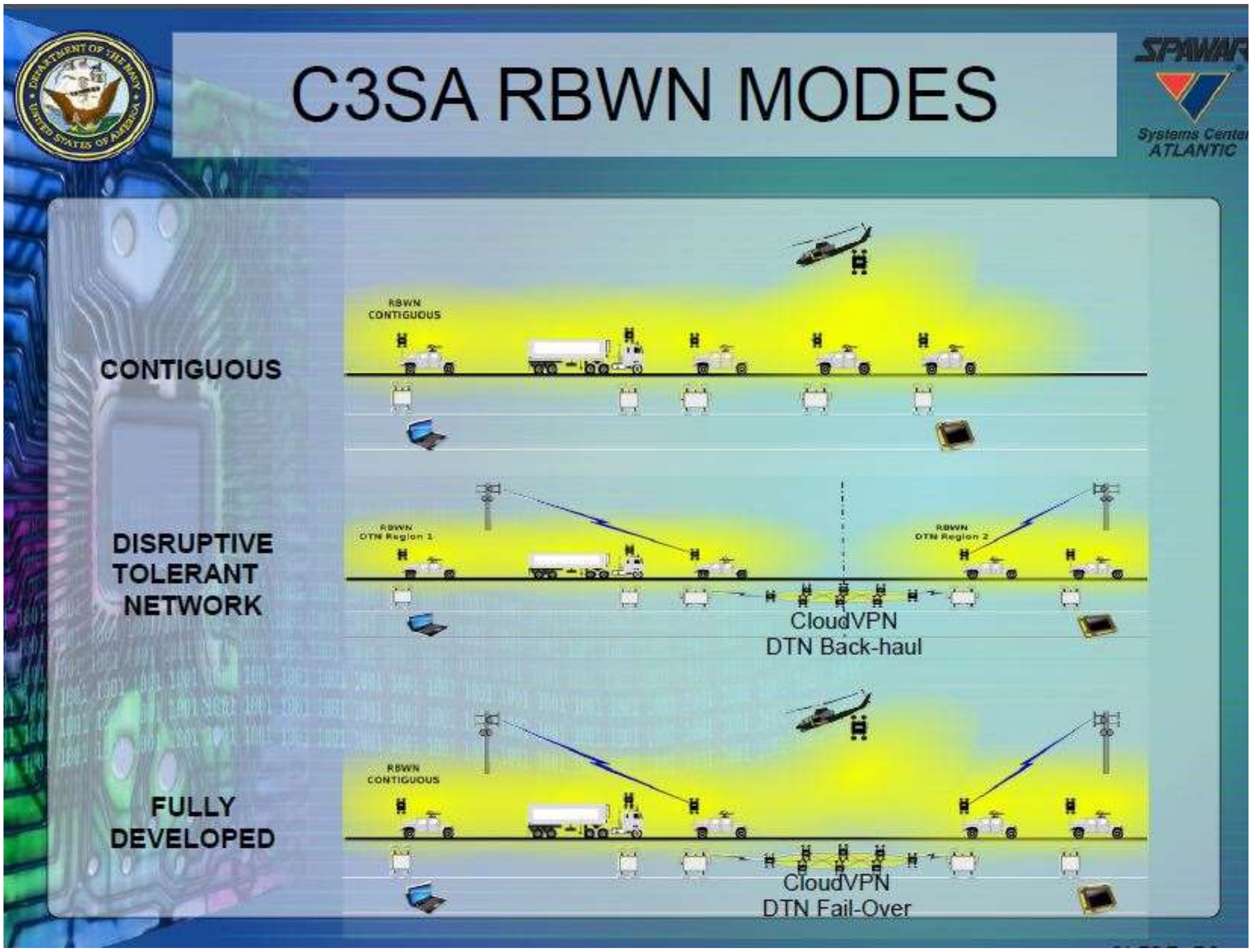
Pub Sub aware Applications on the Propagator nodes serve as aggregation and pruning “hubs”, see Figure 2. Chirps passing from-and-to end devices may be combined with other traffic for forwarding. Applications provide this networking on behalf of devices and integrators at levels "above" and "below" themselves. Any of the standard networking protocols may be used, and propagator nodes will perform important translation functions between different networks (power line or Bluetooth to ZigBee or WiFi, for example), essentially creating small data “flows” from chirp data streams.

Other Trusted applications and agents, many residing inside the Propagators, coordinate the function and control of dumb small, cheap, and copious IOT devices through Software Defined Networking (SDN) paradigms for the edge.

MeshDynamics has been developing an open-source propagator platform for disruption tolerant networking for the US Navy and US Department of Energy. Propagator nodes support User Space Application Layer within an OpenWRT architecture for deep packet inspection, SDN based routing, Video, IFTTT (conditional “If This Then That” rules), etc. These propagator nodes provide autonomous, robust machine control with no assurance of internet connectivity through the built-in applications agents.

The end result is a Publish/Subscribe (Pub Sub) network that can be extended from Big Data servers all the way to the edge of the network while still maintaining a degree of responsive local autonomy. A variety of standards-based SDN protocols may be implemented on the distributed applications agents.

"*MeshDynamics Scalable and Open Pub Sub enables us to rapidly integrate with Enterprise Class, OMG (Object Management Group)-approved, industry-standard messaging systems from RTI (Real-Time Innovations), PRISMTECH, OpenDDS, and others to provide assured real time end to end performance, even if we scale to billions of devices at the edge.”* said Curtis Wright, Sr. Research Systems Engineer, Space and Navy Warfare Center.



C3SA RBWN MODES

CONTIGUOUS

DISRUPTIVE TOLERANT NETWORK

FULLY DEVELOPED

CloudVPN DTN Back-haul

CloudVPN DTN Fail-Over

.

FIGURE 3 Disruption Tolerant, Semi-autonomous networks

Nearly everything described above takes place autonomously and automatically within the Disruption Tolerant Mesh Network (Figure 3). Propagators and their applications even route around failures of links or nodes and deal with issues created by mobility of network elements. But some of the most important capabilities of propagators are distributed application intelligence agents that can allow higher level functions to “tune” the propagator network as part of an overall publication/discover/subscribe infrastructure and/or create application instantiations (e.g. machine controllers) within the nodes themselves. These permit connected devices to continue to operate when the broader network connection is lost.

Propagator nodes also act as traditional switches or routers for IP traffic, while translating and packaging chirp traffic into IP packets for forwarding. Because the propagator nodes incorporate both chirp-based and traditional protocols, they are the natural point of integration for emerging IOT and legacy networks.

Sharp Corporation recently announced the QX-C300 series of networking devices that acts as a traditional WiFi Access Point and also provides connectivity for IOT devices such as cameras to deploy what the company calls “smart networks.”

“MeshDynamics’ propagator node software allows us to deploy WiFi networks today with minimal additional wiring and also incorporate emerging Internet of Things devices on the same infrastructure today and in the future.” said Mr. Arai Yuji, GM, Communication Division, Sharp Electronics, Japan.

Chirps enable discovery

As in Nature, the chirp structure lacks a unique device identification, but does provide a classification of the device type within the public markers. In a hierarchical fashion, devices may be classified as being sensors or actuators, then the type of sensor, and other further defining characteristics, e.g., model number, etc. It is anticipated that there will be an industry-specific open-source registries of chirp identifications that OEM manufacturers utilize and extend.

Individual OEMS may create a new chirp genre or add private extensions to existing chirps to allow more end-to-end capabilities and control. It is expected that a number of industry working groups and SIGs will join together to refine sub-classifications to suit their needs. Importantly, this data structure allows a "start fast and accommodate change" evolutionary approach that will speed deployment of the IOT versus waiting for a conventional standards process. Nature didn't.

Chirps marked with a type ID open a truly powerful opportunity within the IOT. In many cases, an enterprise IOT network may be "closed", using the private markers within the IOT packets to secure the data within. (Chirp data security is discussed in more detail in "What about security?" below.) But in many other cases, individuals and organizations will open their chirp data streams to the public, allowing anyone to make use of the published data. (This is somewhat analogous to the streaming webcams that are made available on the Internet today).

Because these chirp streams are tagged with device type, "interested" integrator functions may "discover" potentially useful chirp streams based on geographic location, device type, or data patterns. Thus, the architecture is *receiver-based*, with integrator functions seeking out and subscribing to data streams of interest.

While the chirp data structure is very different from traditional networking protocols, it will be all that is needed for the majority of sensors, actuators, and devices on the IOT. And type-marked chirp data streams open tremendous opportunity for leveraging the expected tsunami of data.

What about security?

In a chirp-based IOT, huge packets, security at the publisher, and assured delivery of any single message are passé. Chirps instead mirror nature with massive publish and subscribe networks based on "light" pollen. As with nature's pollen, pheromones, and birdsong, many may recognize that there is some data being published, but only the *correct* receiver will have the key to fully unlock the meaning.

Chirp IOT is "female" (receiver-oriented) versus the "male" structure of IP (sender-oriented). When messaging is *receiver-oriented*, networks survive the relentless broadcast storms of spring. IP-based networks would collapse within days.

The security threat of billions of (conventional IP based) IOT devices is very real. IP based messaging (from Server to Device) simply wont scale. IP is a sender-oriented form of messaging – thus, it mandates Encryption. That is a losing battle. Moore's Law is slowing down and in any case Metcalfe's law is exponential e.g. O(n*n). There is a good reason why Nature uses open, extensible, subscription-based (receiver-oriented) "messaging."

Further security is achieved through the applications agents in propagators (see Figure 1). Secure data may be flowing through the propagator node network alongside open data, but is unintelligible without the encryption keys provided to the application agents. This is similar to receiver-oriented schemes found in nature, such as when air transports both proprietary (e.g., pollen) and open "signals" (e.g., birdsong). Individual propagators may be biased to transport or discard secured or open data.

One of the hidden security benefits of the chirp architecture is that there is no end-to-end direct connectivity to individual end devices – the propagator is always in the data path. With the potential for sophisticated security applications within the propagator, end devices are invisible to hackers and vandals. This approach is far simpler and cheaper than managing encryption and security at millions (or billions) of end devices – which further need not be burdened themselves with the processing power and memory needed for security applications. Small, dumb, cheap, copious – *and* secure. Security is obviously a key concern for MeshDynamics Military OEM licensees.

What about standards?

The "Standards conundrum" suffers from the same misleading logic as requiring unique MAC IDs to address an IOT device. I alluded to this fallacy in my book where I describe how there are many John Smiths in the world, but the ones I have in my rolodex are sufficiently distinctive (to me, based on context) to be "uniquely" addressable. Local Uniqueness is enough. Nature concurs. In combination with receiver-oriented messaging, it is even exploited in how prolonged "broadcast" storms of spring disseminate pollen. The winds that carry the pollen are not "global" and time to live is inherently constrained. The same sort of broadcast over IP networks would be crippling, but each propagator effectively and automatically segments its local end devices from the network and vice-versa.

Propagators play a further role in managing standards and accelerating the proliferation of the IOT. Because each may contain applications agents tuned or defined by elements "higher up" in the network, they may serve as a translator for a wide variety of end devices. Chirp-based or IP-based, to name two, but also any variety of ad hoc or standards or proprietary protocols found in older M2M networks. The propagator node effectively isolates and "spoofs" addressing, control timing, and other characteristics of the data stream. Again, addressing need not be globally unique – or even globally *understood* – application agents in the propagators host the necessary intelligence to handle all conversions.

Along with the possibility of a very wide array of physical interfaces on propagator nodes (wired, optical, and wireless, for example), these conversion capabilities ease standards issues and allow rapid migration of legacy networks to the IOT.

Summary

The future world of small, dumb, cheap, and copious sensors, actuators, and devices demands rethinking at both ends of the scale. At the far reaches of the network, simplified chirps will minimize lifetime costs for the myriad end points of the Internet of Things. At the same time, powerful networking and applications tools concentrated in propagator devices will allow unprecedented control and flexibility in creating huge enterprise networks of diverse elements by extending industry-standard Software Defined Networking capabilities. Fully exploiting the power of the Internet of Things will grow from a total rethinking of network architectures.

About the author

The emerging Internet of Things architecture and MeshDynamics wireless mesh networking propagator technology has been influenced by the Robotics and Machine Control background of founder Francis daCosta. (Early MeshDynamics nodes were installed on mobile robots.)

Francis previously founded Advanced Cybernetics Group, providing robot control system software for mission critical applications. These included local and supervisory real time machine-to-machine control. At MITRE, he served as an advisor to the United States Air Force Robotics and Automation Center of Excellence (RACE).

In 2012, Intel sponsored Francis' book *Rethinking the Internet of Things* (Apress, 2013). It was a finalist for the 2014 Dr. Dobbs Jolt Award.