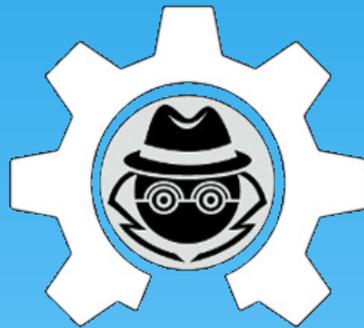


PatternAgents

Open Source Design Patterns for Electronic Systems



Tom Moxon
@PatternAgents

Thank You!

* A Big Thank You to :

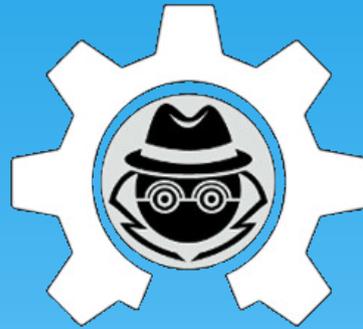
Westmark Electronics, Inc.

for providing the venue for this meet-up!

<http://www.westmarkco.com/>

(Manufacturers Representatives)

Beacons for Bees



- * Applying Geo-Location to Wild and Domestic Bee Hives

Agenda

- Honey Bees / Colony Collapse Disorder
- Hardware for Beehives
- The Physical Web
- Beacons
- Proximity Beacon API
- Hardware for Beacons
- Beacon Reference Designs

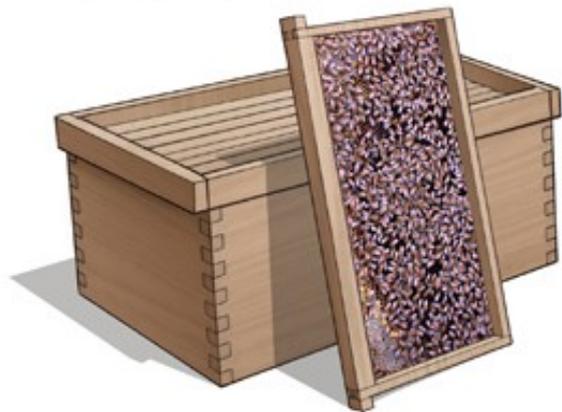
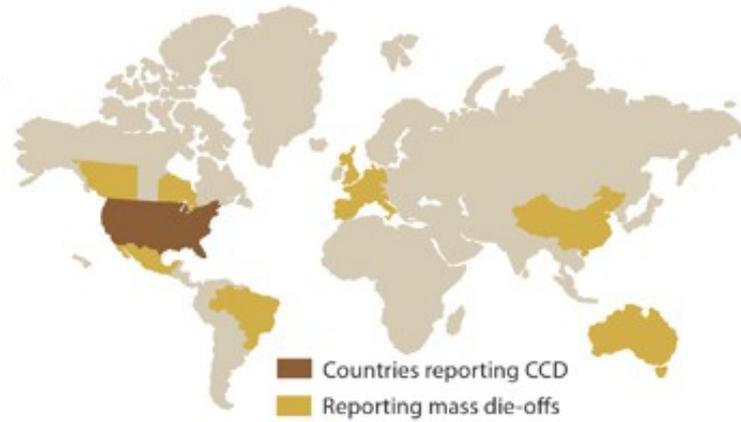


Honey Bee CCD

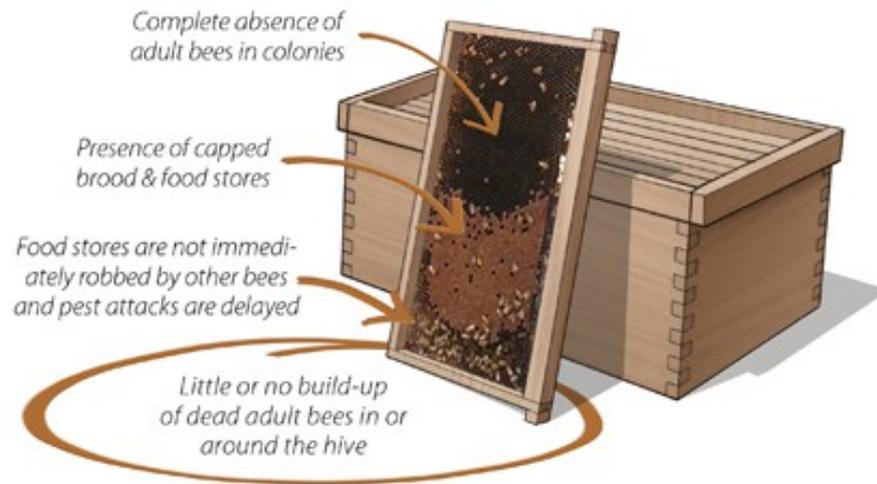
"Never in 40 years had I witnessed the symptoms I was seeing."

- DR. DEWEY CARON, UNIVERSITY OF DELAWARE

What is Colony Collapse Disorder? It's a phenomenon in which worker bees from a hive or colony abruptly disappear. "Stress, immunocompromization, and unusually serious infections by common pathogens and/or otherwise and usually benign organisms, seem to have combined to produce this devastating condition, CCD" (Kevan, 2007).



HEALTHY HIVE



COLLAPSED HIVE

Graphics by Tony Linka

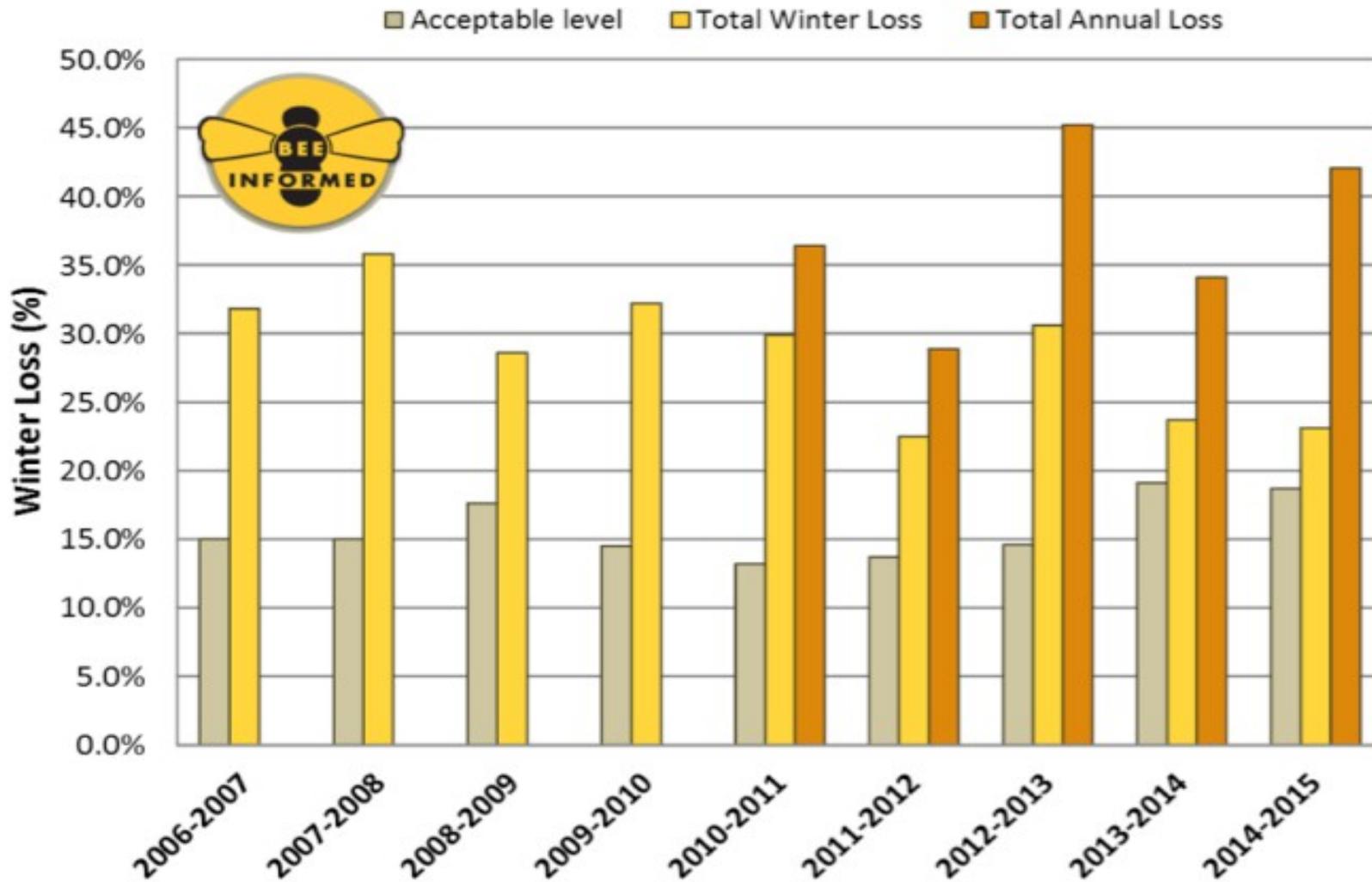
Suspected CCD Causes

There are many causes that have been attributed to Honey Bee Colony Collapse Disorder (CCD), including :

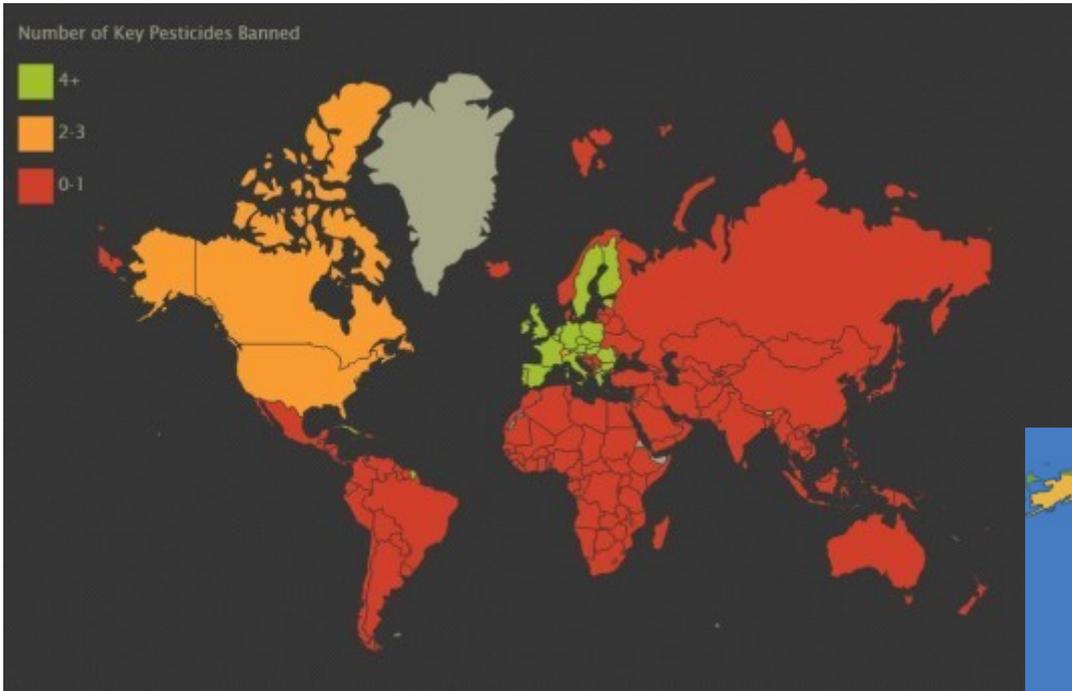
- Wild Habitat Destruction
- Mono Crops (single crops lack diverse food supplies)
- Neonicotinoids and general pesticide overuse
- Mites and other parasites
- Fungal, Viral, or other infections (Nosema, IAPV)
- General Climate Change Issues

U.S. Honey Bee Surveys

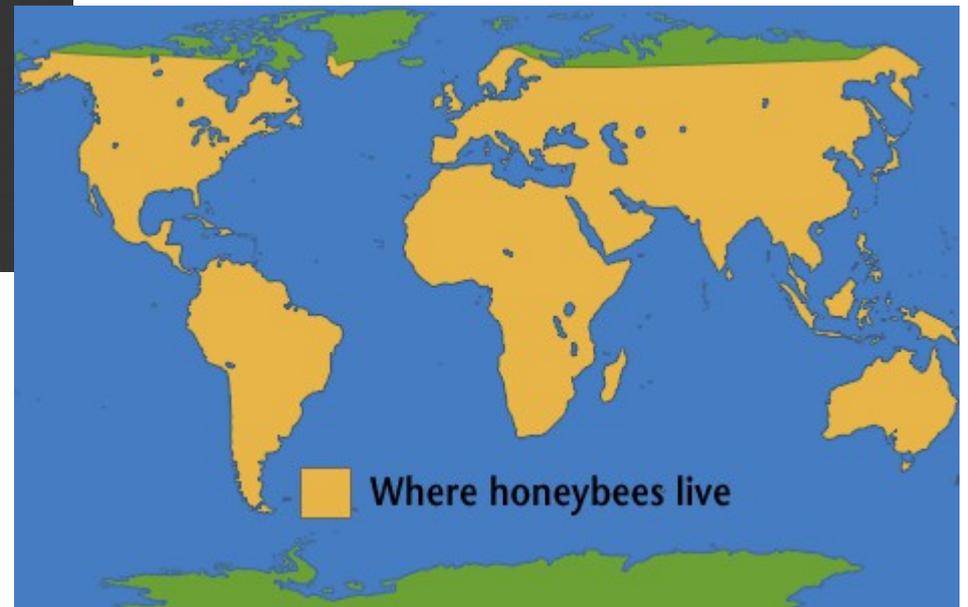
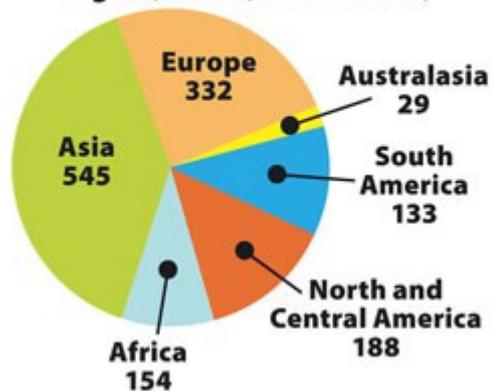
Total US managed honey bee colonies Loss Estimates



Worldwide Impact



World honey production by region, 2005 (1000 tonnes)⁶



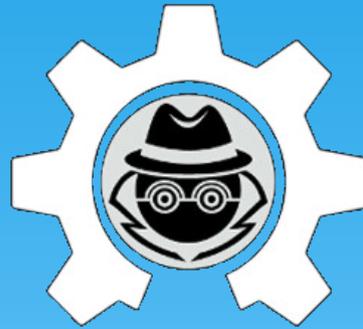
Source: National Geographic

How can we help the Bees?



- Plant Bee Friendly Wildflowers (increase biodiversity)
- Reduce your use of pesticides (companion crops, etc.)
- Raise Awareness of the plight of Honeybees
- Consider your own backyard Hive
- Locate, measure, and monitor wild Hives (little data)

Hardware for Bees

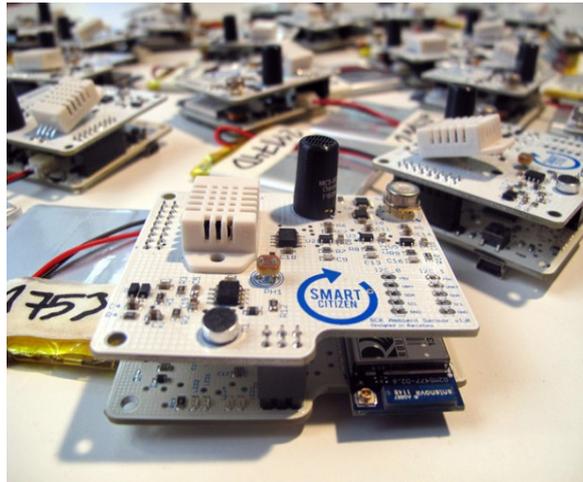


* Working with Hives and Bees

Open Source Beehives



Colorado Top Bar Hive



Smart Citizen Sensors



Barcelona Warre Hive

- * Efforts like Open Source Beehives have been providing easy to build kits and sensors packages for DIY beekeepers

Materials for Hives

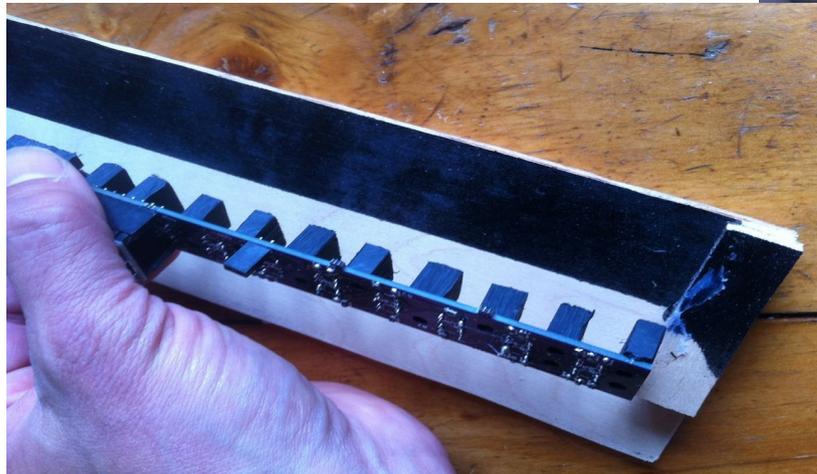
- * Bees are very picky about what goes into the hive
- * Organic materials (wood, bamboo, paper, etc.)
- * FR4 (printed circuit board materials) and conformal coating of FR4 are not welcomed by the bees
(out gassing and aging of materials is a problem)
- * 3-D Printing should use undyed, Poly Lactic Acid (PLA) materials, a corn based, biodegradable, plastic that the bees will generally accept into the hive

Maintenance for Hives

- * Bees may coat circuits, wires, and sensors with propolis, if they don't like the materials
- * If you can coat circuits, wires, and sensors in bee's wax after final test, then the bees will tend to leave them alone
- * If you can't coat it in bee's wax, consider a laser cut wooden or bamboo box, and seal the box edges and openings with bee's wax
- * Replaceable covers for in-hive sensors, paper/gauze

Local Efforts – Bee Counting

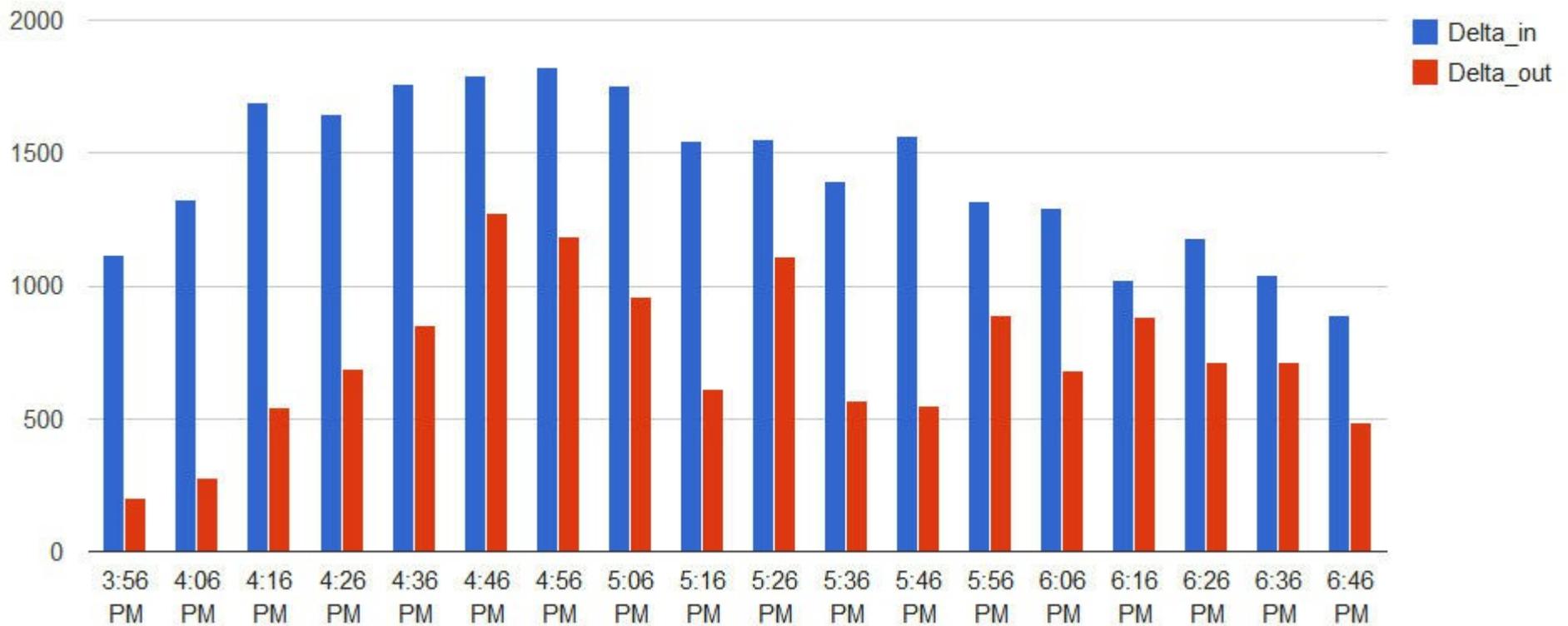
- * Open Source design for optical counter
- * <http://www.instructables.com/id/Honey-Bee-Counter/>



Source: Tom Hudson, PE

Bee Counting

June 21, 2012 - Skidmore Street, 10 minute intervals



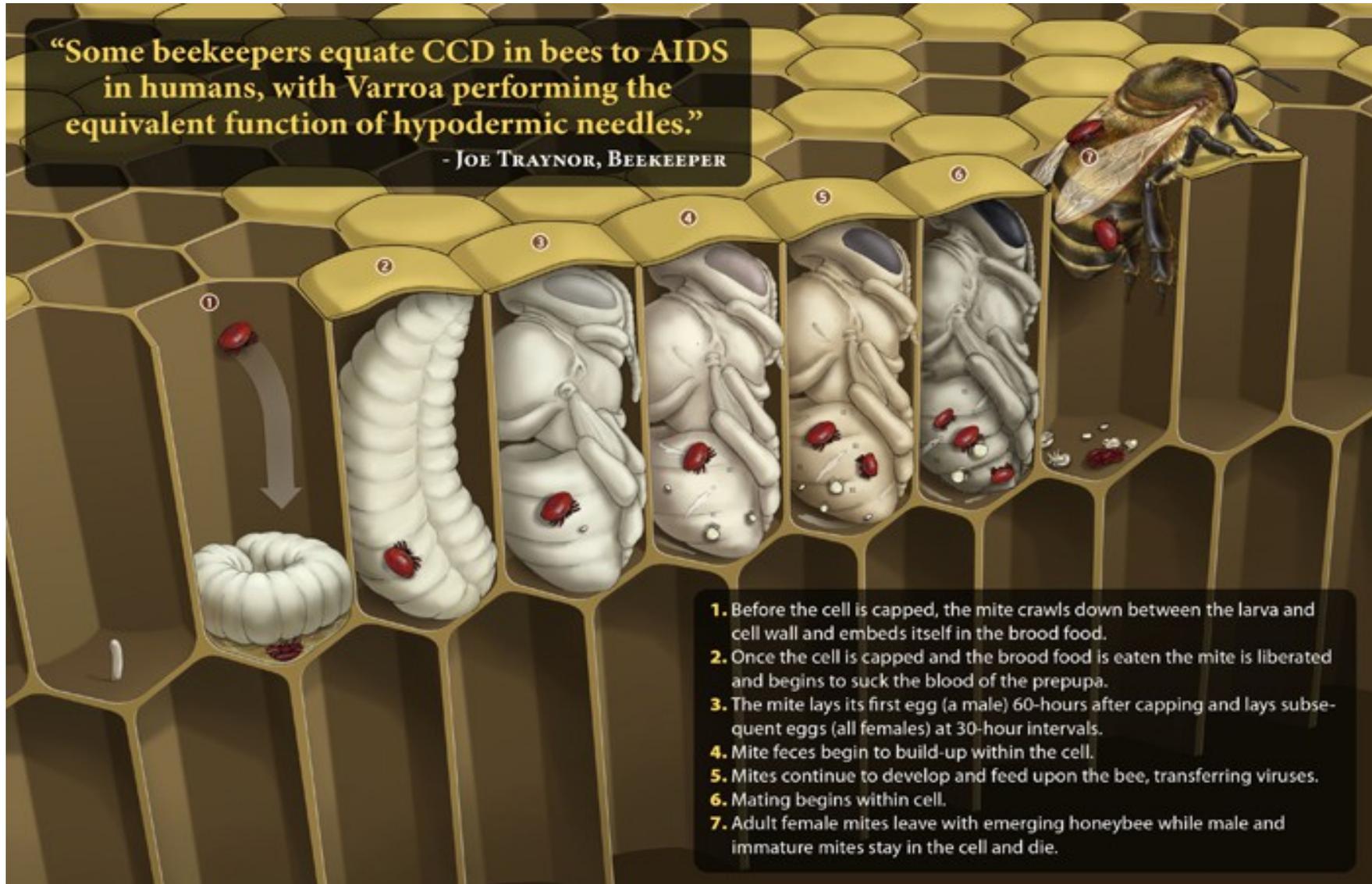
Source: Tom Hudson, PE



Varroa Mite Infestation

“Some beekeepers equate CCD in bees to AIDS in humans, with Varroa performing the equivalent function of hypodermic needles.”

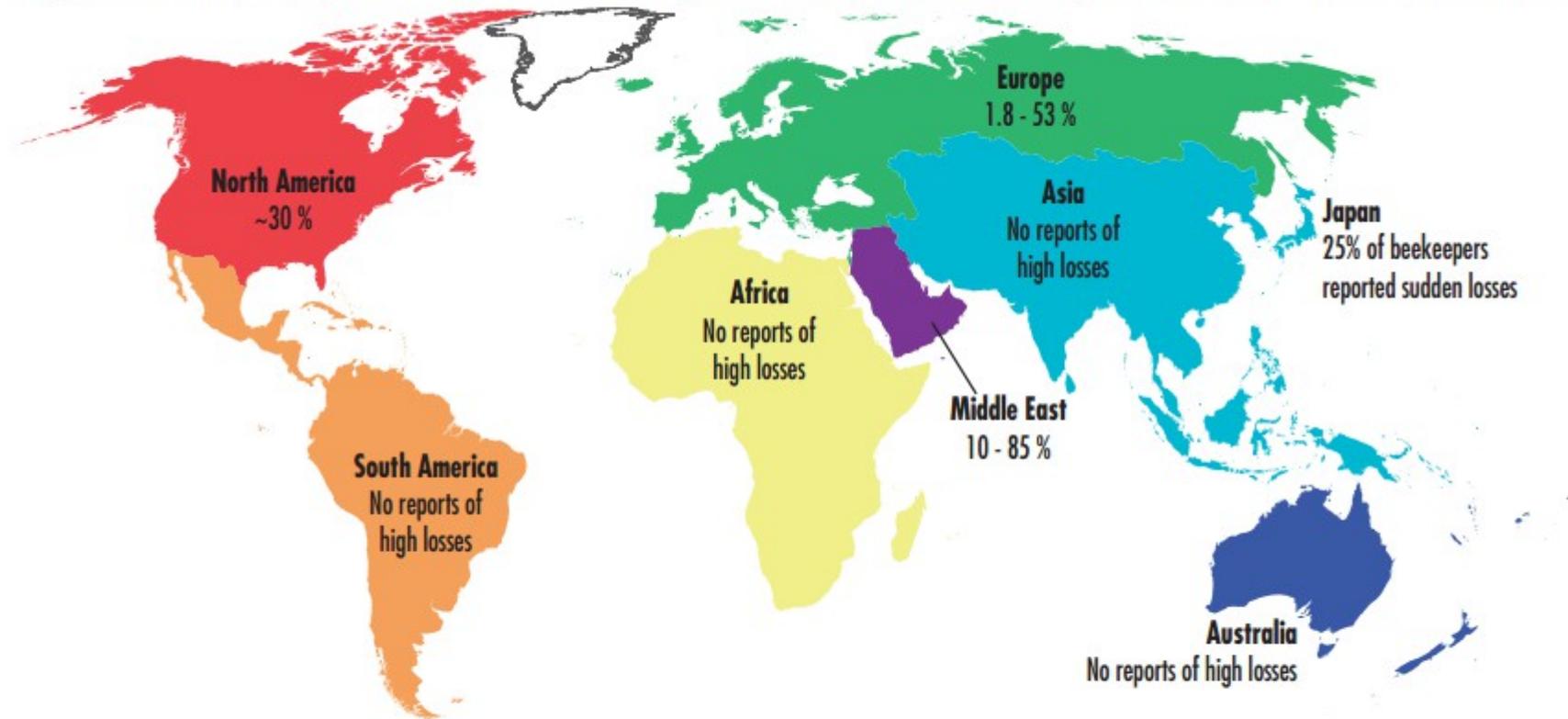
- JOE TRAYNOR, BEEKEEPER



1. Before the cell is capped, the mite crawls down between the larva and cell wall and embeds itself in the brood food.
2. Once the cell is capped and the brood food is eaten the mite is liberated and begins to suck the blood of the prepupa.
3. The mite lays its first egg (a male) 60-hours after capping and lays subsequent eggs (all females) at 30-hour intervals.
4. Mite feces begin to build-up within the cell.
5. Mites continue to develop and feed upon the bee, transferring viruses.
6. Mating begins within cell.
7. Adult female mites leave with emerging honeybee while male and immature mites stay in the cell and die.

Varroa Mite Losses

Figure 8. Yearly average of managed honey bee colony losses due to Varroa mite



*The Varroa mite, *Varroa destructor*, is one of the threats to managed honey bee colonies. Elevated colony losses reported from the USA, Europe, the Middle East and Japan are related to high mite infection³⁹.*

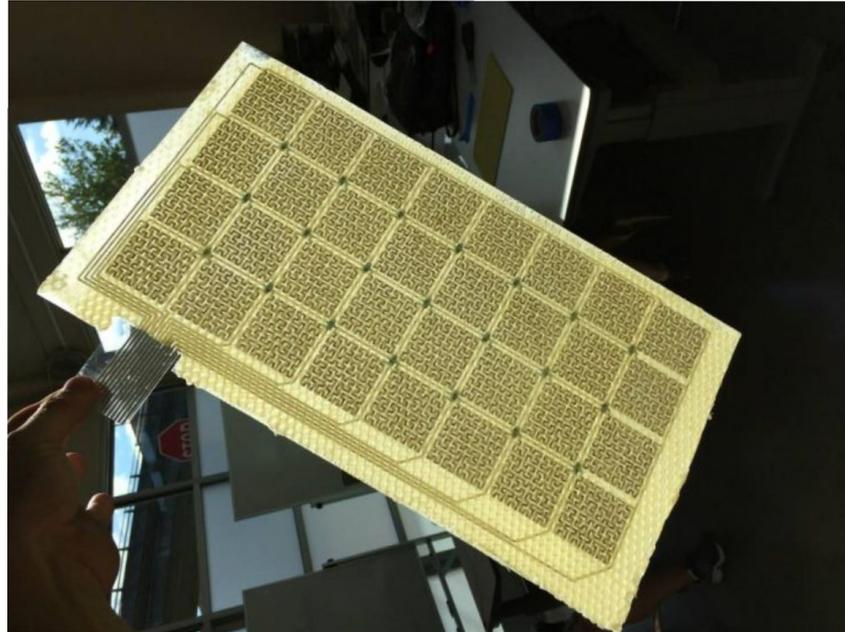
Controlling Varroa Mites

- * Killing the Mites without killing the Bees is difficult
- * Better to interrupt their life-cycle by sterilization
- * Elevated Temperatures in select areas of the hive
- * Existing & patented solutions not working well...
(brute force, difficult to implement, genetic impact)

Circuit Boards for the Bees



- * Eltopia has been developing and testing new control methods for Varroa Mite and other pests



Hive Monitoring Resources

Intelligent Beehives

Open Energy Monitor

HiveTool

Colonymonitoring.com

Hackerbee

Honey Bee Counter

Luke Aldridge Rice

Intelligent Beehives at Yale

Arnia

The Physical Web



- * The physical web is about extending the power of the internet to everyday physical objects and places. The goal is to be able to walk up to any “smart” physical object (like a parking meter, vending machine, etc.) and be able to interact with it without having to first to download an “app” (application program).

The Physical Web



- * You've all seen the QR-Code Symbols on posters or stickers, next to a store, on a product display, on a bus stop sign.
- * It holds an “encoded URL” or INTERNET address, using a smartphone camera and application to “SCAN” the QR-Code, and launch the web browser to that address.
- * Can yield a more seamless “physical” to “virtual” user experience

Near Field Communication



- * NFC – Near Field Communication
- * “Tap” to Join/Pair/Exchange

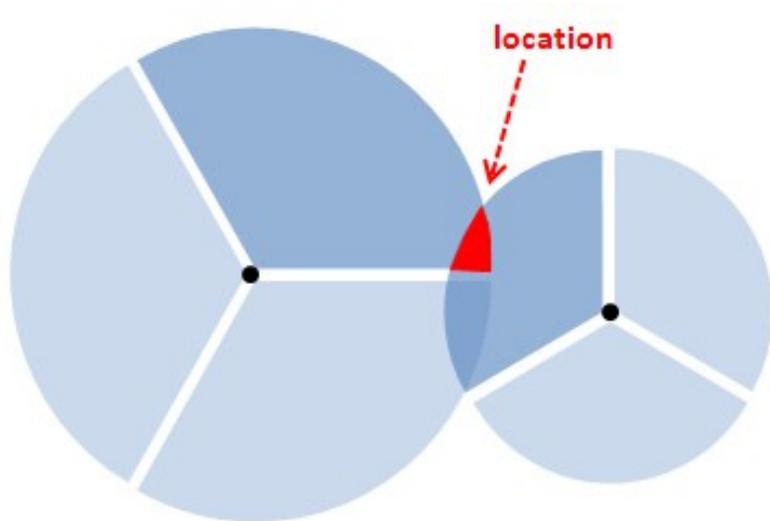


Global Positioning System



- * Most smart phones include GPS capability these days
- * Uses a LOT of battery power
- * Doesn't work well indoors or under the forest canopy
- * Can give fine grain position (3meters)

Cell Tower Positioning



Cell phone detected within a certain distance of two cell towers with directional antennae.

- * Triangulation/Location
- * Not very accurate
- * Uses cell phone
- * Only where cell towers are...

Indoor Positioning System



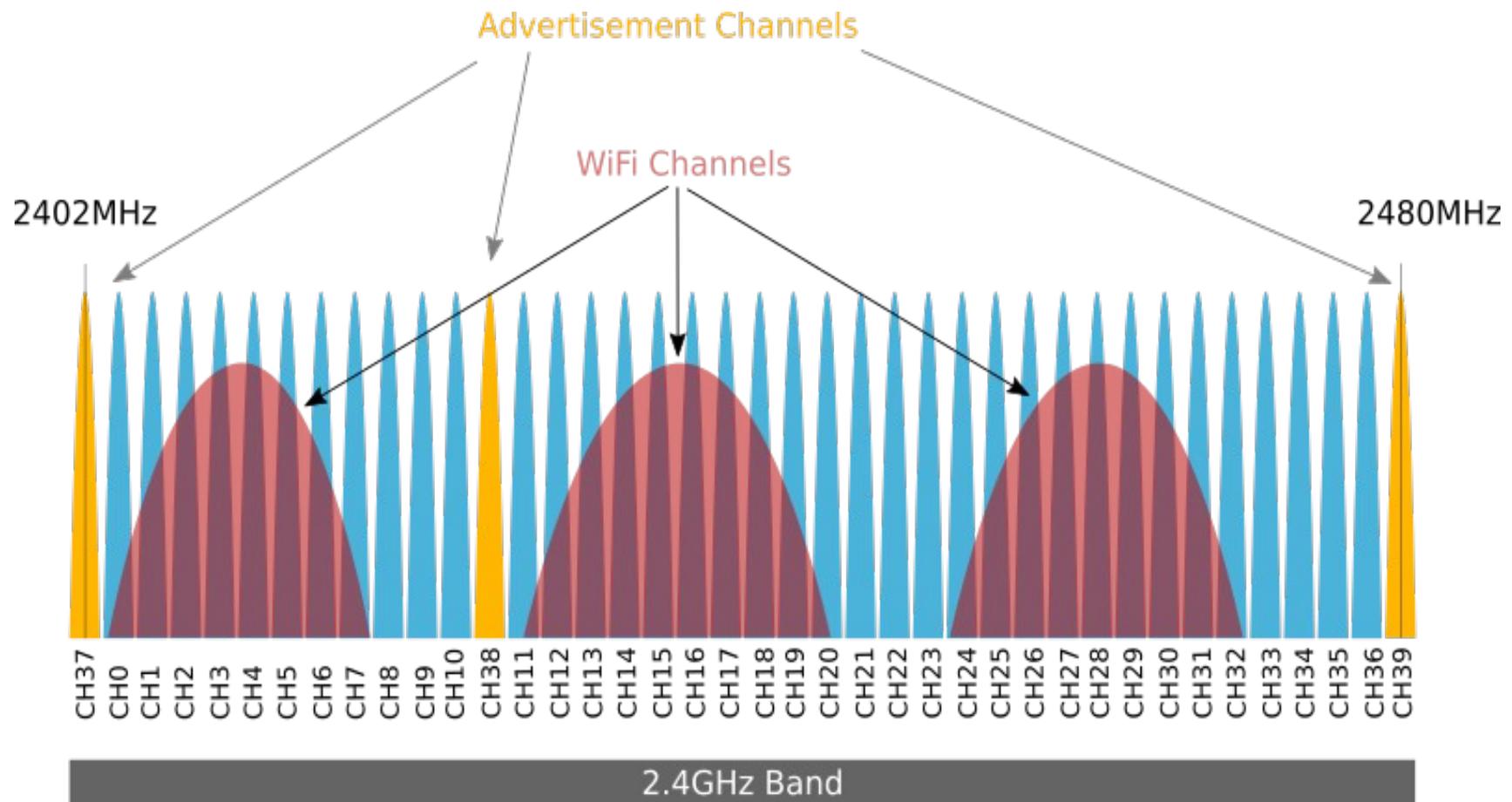
- * Uses Wi-Fi signal strength (RSSI)
- * Triangulation from known location of several Wi-Fi access points
- * Popular in Malls, Campus, etc.
- * Needs Wi-Fi radio on (uses power)

Bluetooth Communication



- * A Wireless Personal Area Network
- * Originally Nokia “Wibree” (2006)
- * Merged into Bluetooth SIG (2010)
- * Very low power transmitter
- * Ubiquitous now...

Bluetooth vs. WiFi



Okay, so what is a Beacon?

Imagine that each Lighthouse could flash it's light in a unique pattern that would identify each lighthouse individually...



- * A Bluetooth Beacon functions similarly to a Lighthouse, it transmits information about a location, using a Bluetooth Low Energy radio, programmed to emit a known pattern, or universally unique identifier.

(i.e. “You are near here!”)

Lighthouse Photo by
Dennis Jarvis

Why Beacons for Bees?

- * Raise awareness of pollinator health problems
- * Proximity warning for those allergic to Bees, Wasps, etc.
- * Locate, monitor, and protect Wild Hives
- * New approach to Geo-Tagging wildlife habitat
- * More appropriate technology for remote areas
- * Lower power than existing monitoring solutions (Wi-Fi uses more power and transmits longer)

What is an iBeacon?



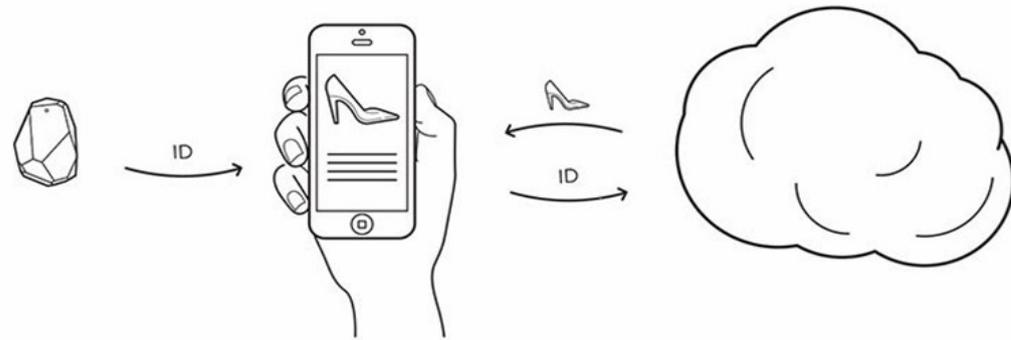
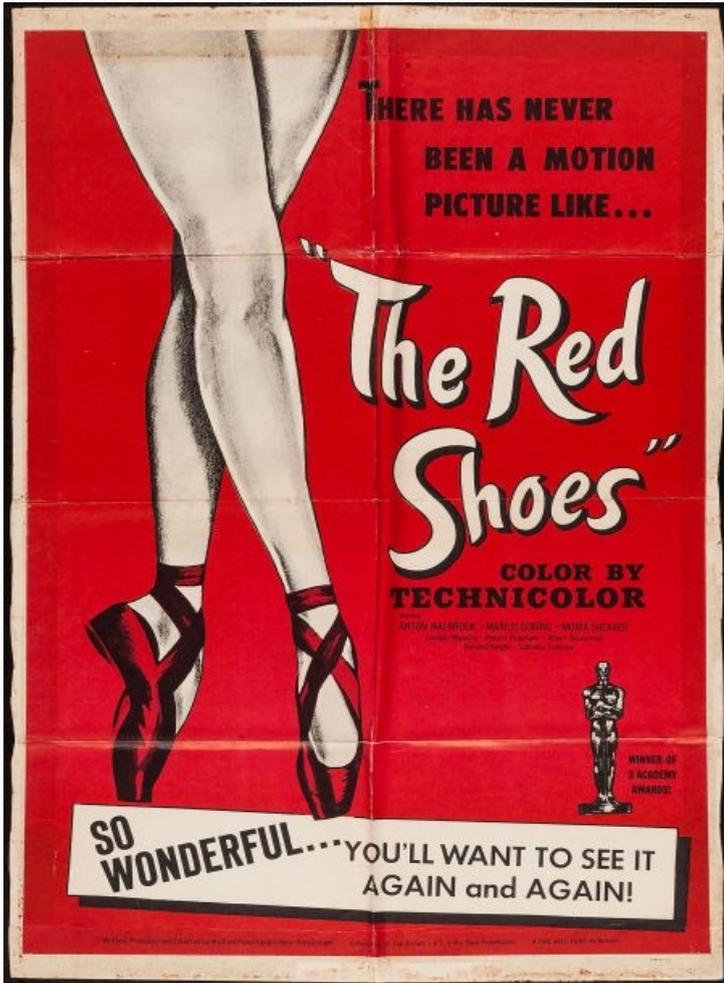
- * A proprietary and licensed Beacon system developed by Apple in 2013 to create location aware services for Apple Pay.
- * iBeacon uses Bluetooth low energy proximity sensing to transmit a universally unique identifier (UUID), which is picked up by a compatible application or operating system.

The unique identifier can be used to determine the device's physical location, track customers, or trigger a location-based action on the device such as a check-in on social media, or a push notification.

First Use Case

- * Originally, Apple envisioned the iBeacons to be run on iPad Tablets used as Point-of-Sale Systems.
- * When you are close enough to the Point-of-Sale System, a coupon or membership offer might appear in the application (“the app”) running on the users iPhone
- * Didn't need any dedicated hardware, as the Bluetooth Radios were already present in the iPhones and iPad tablets

How iBeacons Work



Source: estimote.com

iBeacon - First

- * First Beacon Specification/Most widely installed base
- * Requires the Implementer to develop an “App”
- * Requires the User to install that “App”
- * Uses Vendor/Manufacturer Specific BT Frame
- * “One Beacon, One App”
- * Hard Codes UUID to Application (Maintenance...)

iBeacon – Apple Pay

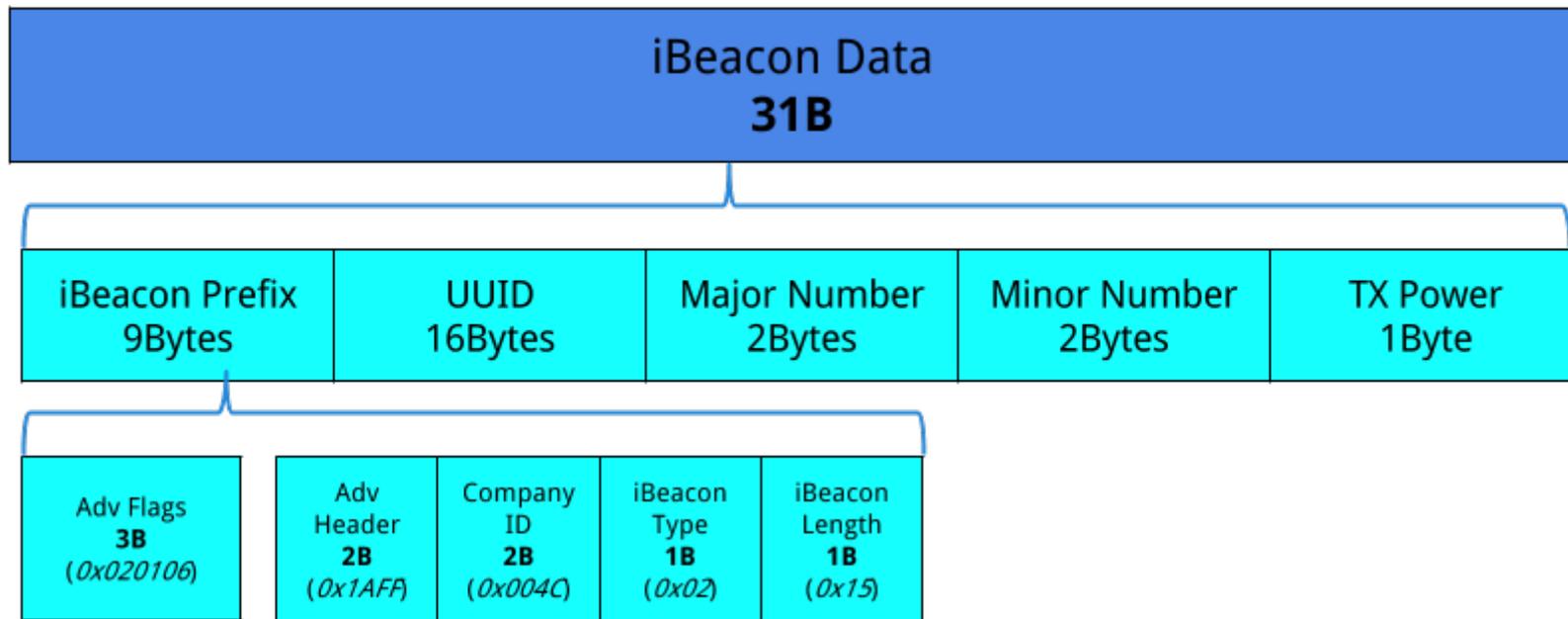
- * Can be very costly to implement and maintain
- * Costs : \$10K typical startup (small event)
: > \$250-\$500K (large event)
: Apple Passbook can be less expensive
- * Users : Apple, Sephora, Dunkin' Donuts,

Potential complementary usage of iBeacon/BLE and NFC



Source: Apple Computer

Advertising Packet iBeacon



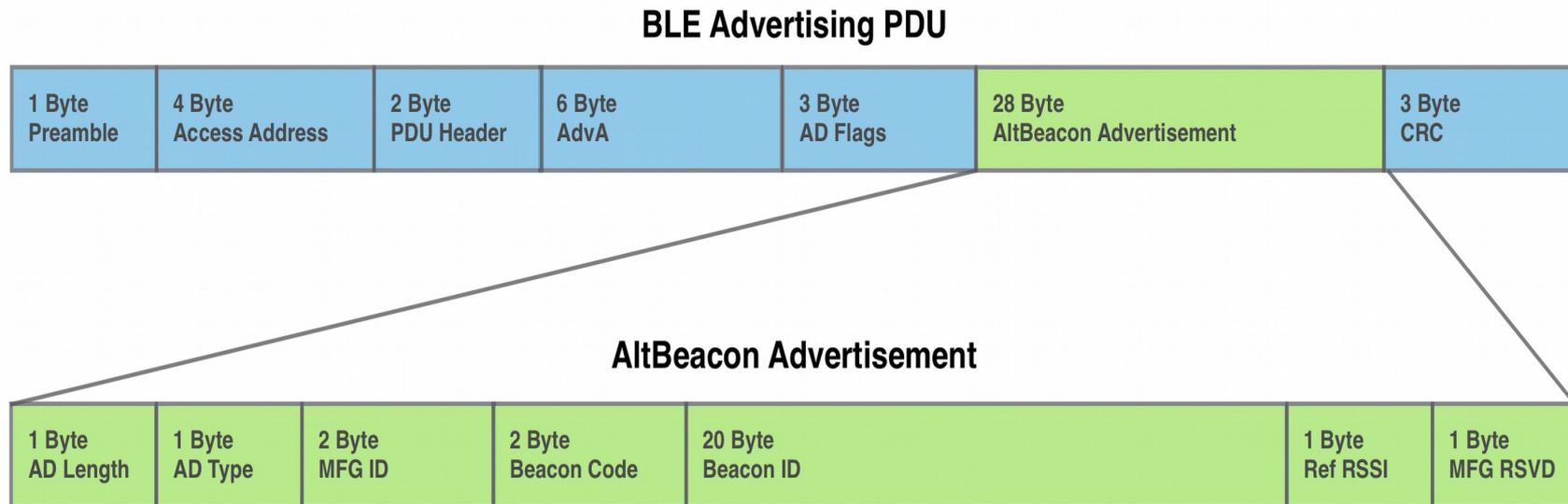
- * Custom Prefix (Apple Specific) 10 Packets per Second (100ms)
- * (UUID) Universal Unique Identifier is an assigned number and needs to be interpreted by the application to be useful
- * Only 31 Bytes!

What is an AltBeacon?



- * Developed as an Open Source alternative to Apple's closed and proprietary iBeacon technology.
- * Specification by Radius Networks
- * Adds Manufacturer ID, Beacon Code, Reserved Code
- * Started gaining steam until EddyStone Beacons were introduced, now losing share to EddyStone

AltBeacon Packet



- * Interesting Use Cases with Manufacturer ID, and MFG reserved codes
- * Limited installed base

What is Eddystone?

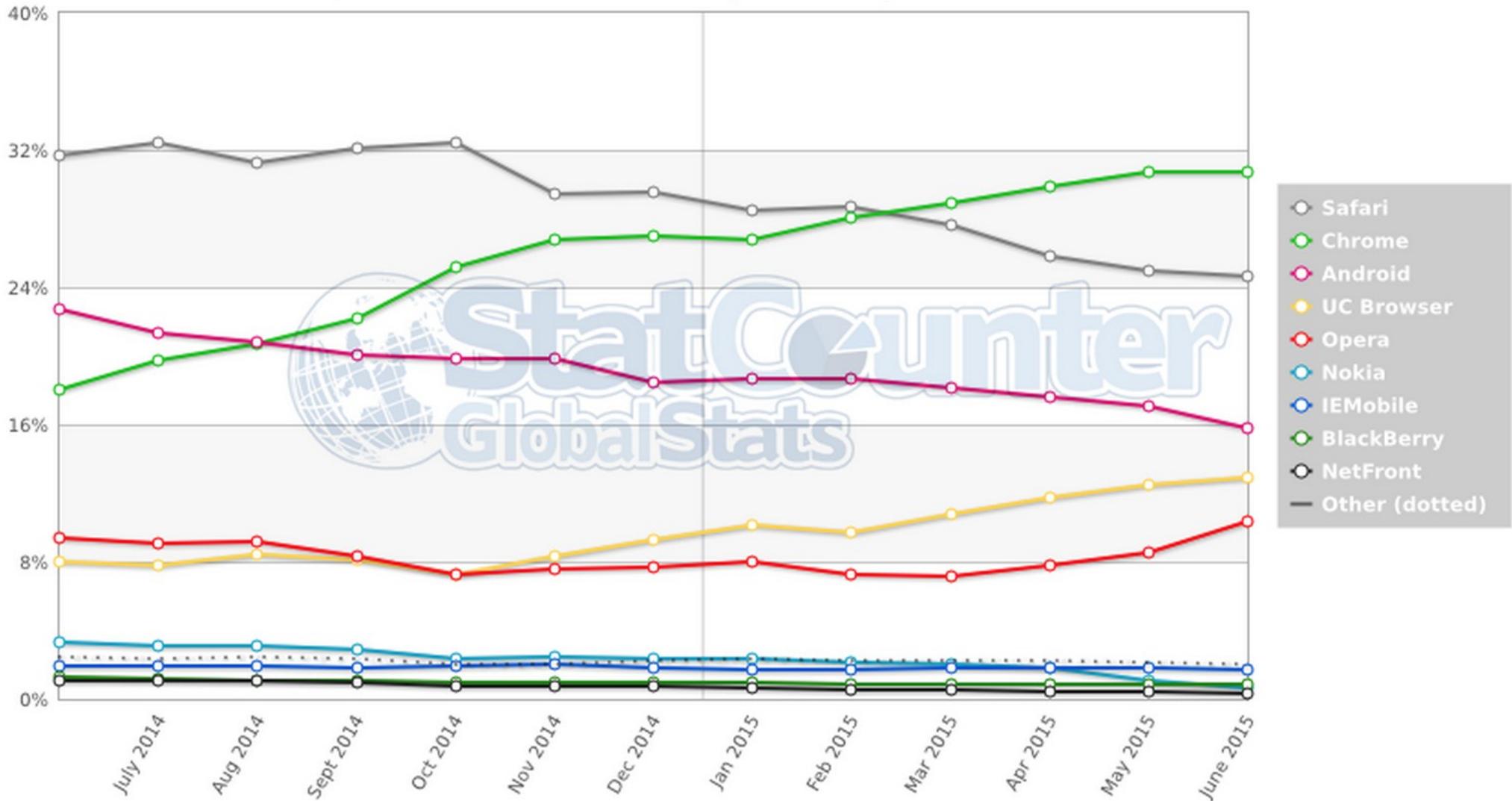


Eddystone

- * Eddystone is an open beacon format developed by Google and designed with transparency and robustness in mind
- * Eddystone can be detected by both Android and iOS devices
- * Eddystone doesn't require a dedicated “app” to function, as it can deliver a target URL directly
- * Google Chrome *IS* the “app”!

Chrome App Share

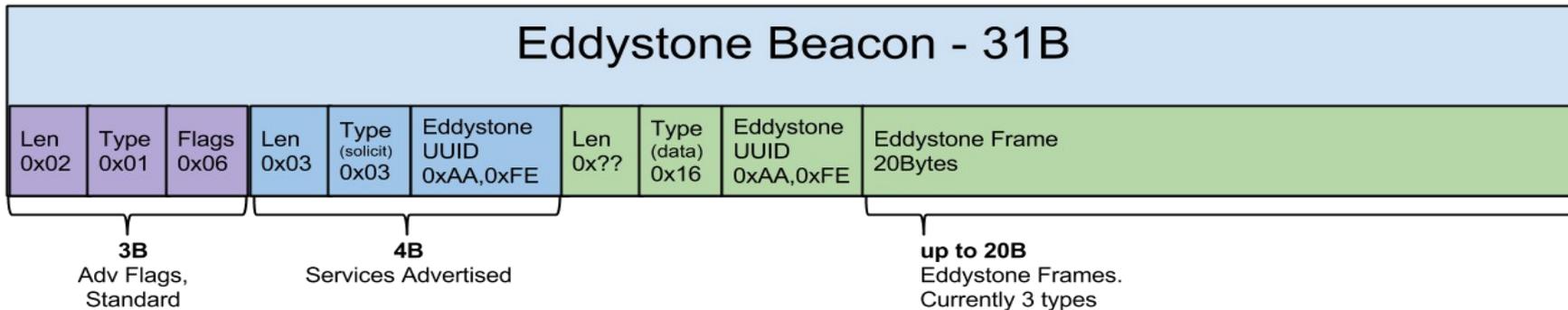
StatCounter Global Stats
Top 9 Mobile & Tablet Browsers from June 2014 to June 2015



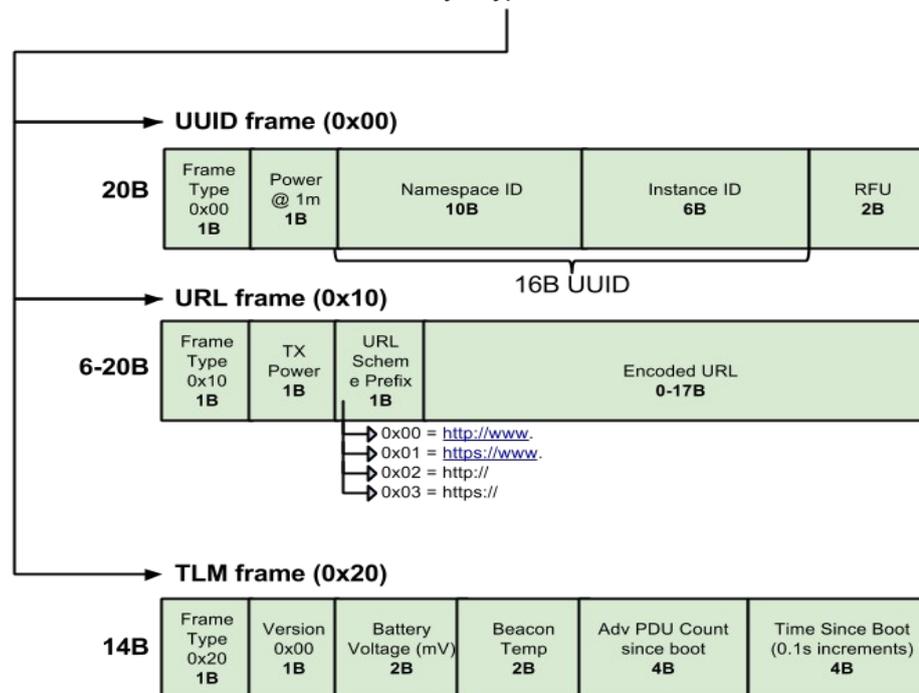
Eddystone Frame Types

- * **Eddystone-URL** : delivers a “short” URL directly to the user when they are in range
- * **Eddystone-UID** : delivers a UUID, similar to how iBeacon works, but 16 bytes in length
- * **Eddystone-TLM** : delivers Beacon Telemetry data for Beacon management (battery level, uptime, temperature, etc.)

Eddystone Packet



- Uses Service Frames
- 1-2 Packets per Second
- 10X less than iBeacon
- Much longer battery life
- Includes Telemetry packet
- No Licensing Required



Eddystone-UUID

- * Functions similar to iBeacon, AltBeacon (app required)
- * Beacon UUID is registered with Google Beacon Dashboard
- * Available for Google Developer Project once registered
- * Open Standard, no licensing, fees, Apple taxes...
- * Use free “Beacon Tools” app to register beacons
- * Almost NO barrier to entry (low cost, open source, etc.)
- * “One Beacon, Many Apps...”

Eddystone-UUID

- * Recommended Practice

- * Use a SHA1 Hash of your company FQDN

`sha1(www.patternagents.com) =
fba4fd8d86e4726632962dfff9f21eafbbe93ed2`

- * UUID = `fba4fd8d86e472663296`

- * Use sub-domains if you need more than one

`foo.patternagents.com`

Eddystone-URL

- * The Eddystone Beacon provides an encoded URL
- * The URL is 17 bytes long (max), and uses an encoded format (UriBeacon format) :

Prefix

0x00 – http://www.

0x01 – https://www.

0x02 – http://

0x03 – https://

Suffix

0x00 - .com/

0x01 - .org/

0x02 - .edu/

0x03 - .net/

- * Works great with a URL shortener, like bit.ly, git.io, etc.
- * Must be an “https://” URL for Google to index it...

Eddystone-URL

- * Eddystone-URL beacons are NOT registered, unless they also transmit UUID frames (which ARE registered)
- * Any HTTPS address can be used and will give a pop-up notification, if notifications are enabled
- * Gives control back to the user/implementer, you can direct the user to any valid, secure URL and they can opt-in/visit
- * No tracking generally, unless the user has a login/account with the referred URL, can be redirected to user page...

Eddystone-URL

- * IMHO, the costs associated with developing apps for iBeacon and AltBeacon are prohibitive unless your company (or application/event/etc.) is large enough to warrant it
- * Eddystone-URL is an equalizer here, it is a much lower cost solution to implement and deploy, and there are no application development costs or Apple licensing costs to incur.
- * Eddystone-URL makes the design, and deployment of location aware services, faster, easier, and less expensive

Eddystone TLM

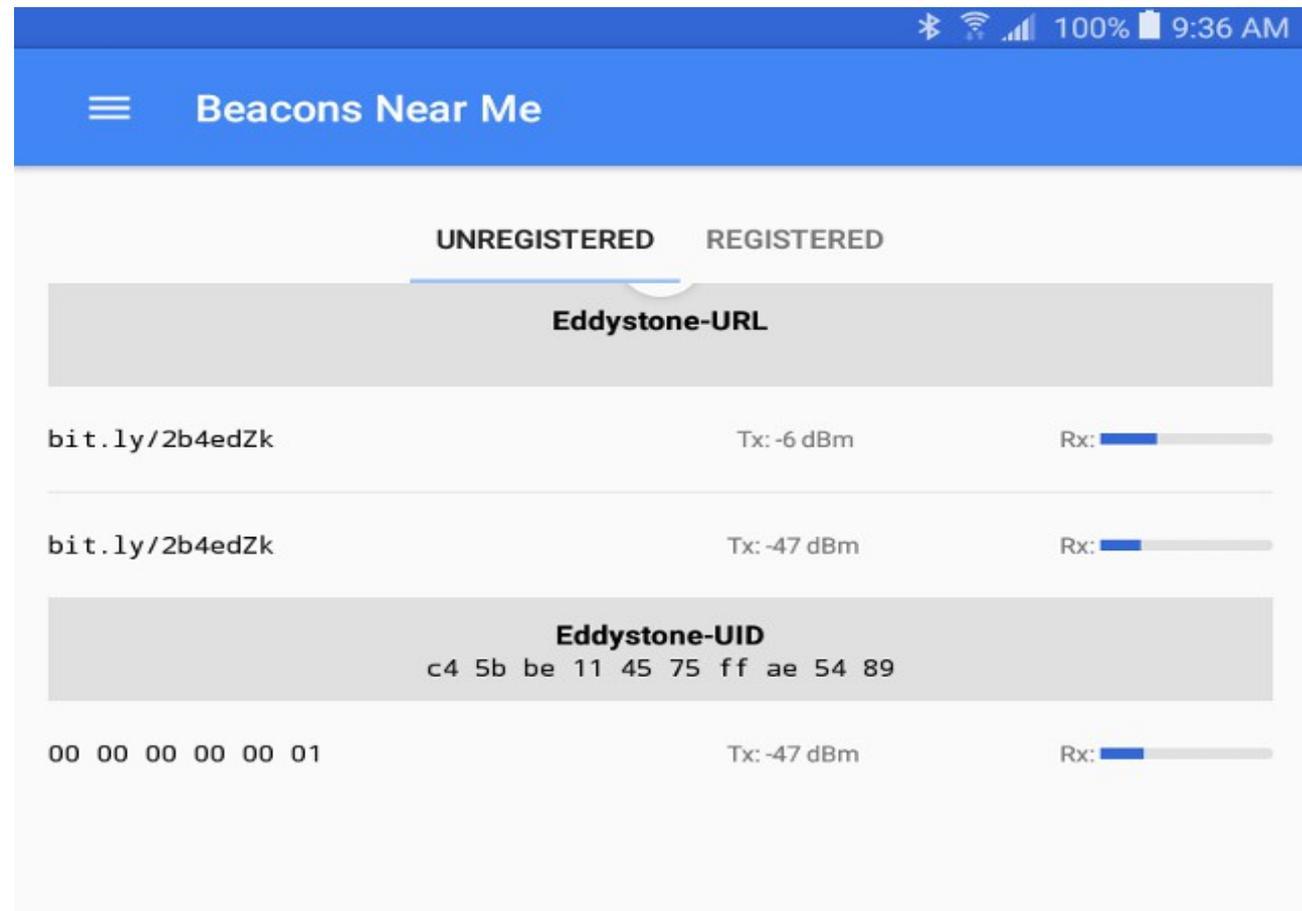
- * Telemetry Frame
- * Great for Beacon fleet management
- * Battery Level, Temperature, Packet Count, etc.
- * Interleaved with other frames occasionally
- * Can be uploaded to cloud via your “app”
(is it stealing my data? Paranoia runs deep...)

Eddystone EID

- * Empheral ID, changes every few minutes
- * A secure key is exchanged during beacon registry
- * Meant to be more secure, not easily spoofed
- * Useful for wearable, medical data devices

Google Beacon Tools App

* Install from Google Play Store



Source: Google

Register Your Beacons

Registering Beacon REGISTER BEACON

Location

Expected stability

Description

Floor level

Properties

Attachments

Enter Location

100% 9:46 AM

REGISTER BEACON

Location



Map data ©2016 Google

[Coordinates 45.513988, -122.648387](#)

Expected stability

Description

Floor level

Properties

Attachments

Stability

- * Mark your beacon as stable, moving, etc.

The screenshot shows a mobile application interface for registering a beacon. At the top, there is a blue header with a hamburger menu icon, the text 'Registering Beacon', and a 'REGISTER BEACON' button. Below the header, there is a 'Location' section with a toggle switch that is turned on. The main content area features a map from Google Maps with a red location pin placed on SE 17th Ave. Below the map, the coordinates are displayed as 'Coordinates 45.513917, -122.648402'. There are four more sections, each with a toggle switch that is turned on: 'Expected stability' with the value 'STABLE', 'Description' with the text 'Patternagents: Open Source Design Patterns', 'Floor level' with the value '1', and 'Properties' and 'Attachments' which have plus signs next to them. The bottom of the screen shows the 'REGISTER BEACON' button again.

Registered



Bluetooth, Wi-Fi, Cellular, 100% battery, 9:48 AM



Beacons Near Me

UNREGISTERED

REGISTERED

Eddystone-UID

c4 5b be 11 45 75 ff ae 54 89

00 00 00 00 00 01

Tx: -47 dBm

Rx: 

Beacon Manager



The screenshot shows the Beacon Manager mobile application interface. At the top, there is a dark blue header with a hamburger menu icon on the left, the text "Beacon Manager" in the center, and a vertical ellipsis icon on the right. The status bar at the very top shows various icons (location, Bluetooth, Wi-Fi, cellular signal) and the text "100% 9:49 AM".

The main content area is divided into two sections, each representing a beacon. Each section starts with a signal strength icon (three curved lines) and a signal strength value.

Beacon 1 (Signal: -84):

- Eddystone URL**
URL: <https://bit.ly/2b4edZk>
- Eddystone TLM**
Battery: 3377 mV
Packets: 130880
Temperature: 25.0
Active since: 2016-09-08 13:11:04

Beacon 2 (Signal: -82):

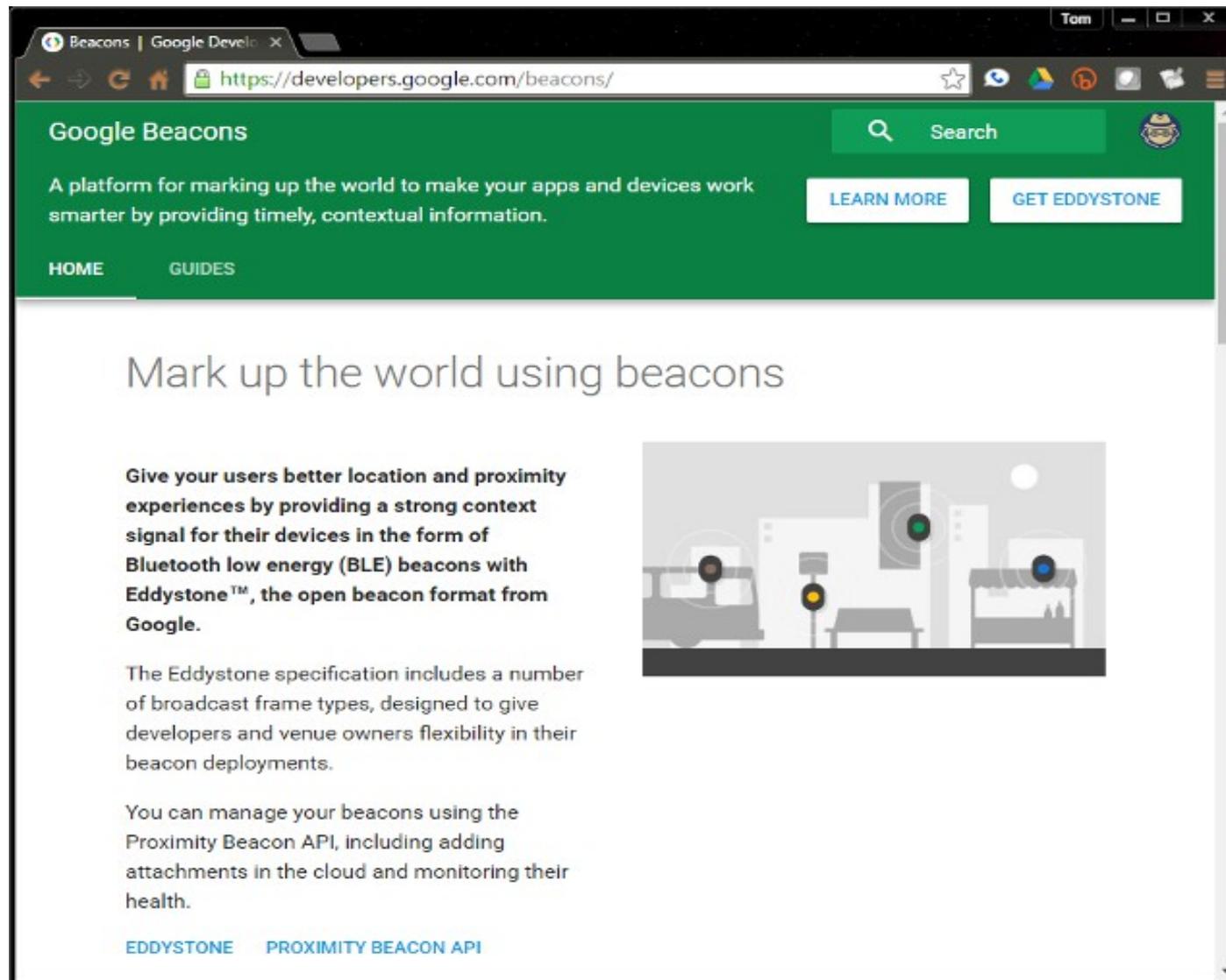
- PatternAgent**
- Eddystone UID**
Namespace ID: c45bbe114575ffae5489
Instance ID: 000000000001
- Eddystone URL**
URL: <https://bit.ly/2b4edZk>
- Eddystone TLM**
Battery: 0 mV
Packets: 5841
Temperature: -127.0
Active since: 2016-09-09 09:30:29

Beacon Toy



- * Turn your phone into a Beacon!
- * Great for impromptu testing, etc.
- * Free app

Google Developer - Beacons



The screenshot shows a web browser window with the URL <https://developers.google.com/beacons/>. The page has a green header with the title "Google Beacons" and a search bar. Below the header, there is a navigation menu with "HOME" and "GUIDES". The main content area features the heading "Mark up the world using beacons" and a paragraph explaining that beacons provide location and proximity experiences using Bluetooth low energy (BLE) with Eddystone™. To the right of the text is an illustration of a city street with beacons on a car, a traffic light, and a building. At the bottom, there are links for "EDDYSTONE" and "PROXIMITY BEACON API".

Google Beacons

A platform for marking up the world to make your apps and devices work smarter by providing timely, contextual information.

LEARN MORE GET EDDYSTONE

HOME GUIDES

Mark up the world using beacons

Give your users better location and proximity experiences by providing a strong context signal for their devices in the form of Bluetooth low energy (BLE) beacons with Eddystone™, the open beacon format from Google.

The Eddystone specification includes a number of broadcast frame types, designed to give developers and venue owners flexibility in their beacon deployments.

You can manage your beacons using the Proximity Beacon API, including adding attachments in the cloud and monitoring their health.

[EDDYSTONE](#) [PROXIMITY BEACON API](#)

Google Beacon Dashboard

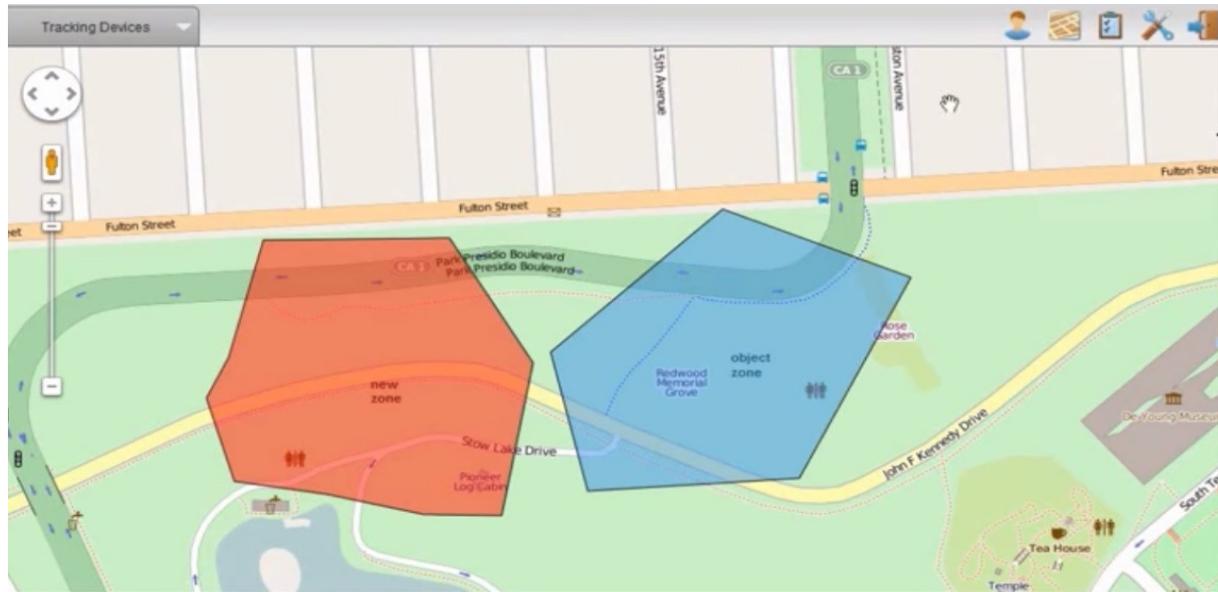
The screenshot shows the Google Beacons Dashboard in a web browser. The browser's address bar displays the URL <https://developers.google.com/beacons/dashboard/#/beacons?pn=10>. The page header is green and contains the text "Google Beacons > Dashboard" on the left, a search bar with the text "Search" in the middle, and a user profile icon on the right. Below the header, the page is titled "Beacons" and "patternagents" with a pencil icon. There is a search input field with the placeholder text "Add search term" and a question mark icon. A green circular icon with a white book symbol is displayed. Below the icon, it says "1 beacon." with a download icon. A table with the following data is shown:

Description	Level	Status
Patternagents: Open Source Design Patterns	1	Active

Proximity Beacon API

- * Eddystone is much more than just a Beacon Format
- * Registered Beacons can have additional Properties and Attachments that get delivered to the app by the Proximity Beacon API
- * Also supports iBeacon, AltBeacon
- * Attachment can indicate functional location (i.e. Store X Front Door, Store Y Aisle 19, etc.)
Makes it easier to replace or retask a beacon
- * Integrates with other Google APIs, Nearby, etc.

What is a Geo-Fence?

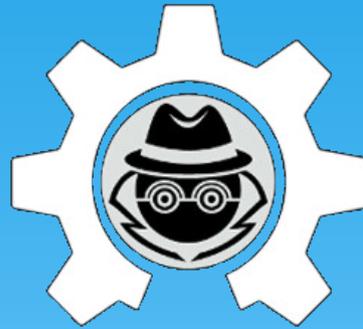


- * A Geo-Fence is a virtual perimeter for a real-world geographic area.
- * Geo-Fences can be implemented using Cell Tower Positioning, GPS location, Wi-Fi Signal Strength measurement or any combination of those.

eBeacons vs. Geo-Fences

- * Chrome is the App
- * Self-Contained
- * User enables Phy. Web
- * Typical 70m area max
- * Best for small areas
- * Needs dedicated App
- * Needs GPS, Cell, Wi-Fi
- * User must Opt-In
- * Typical 100m area min
- * Best for large areas

Hardware for Beacons



* A quick How-To Overview...

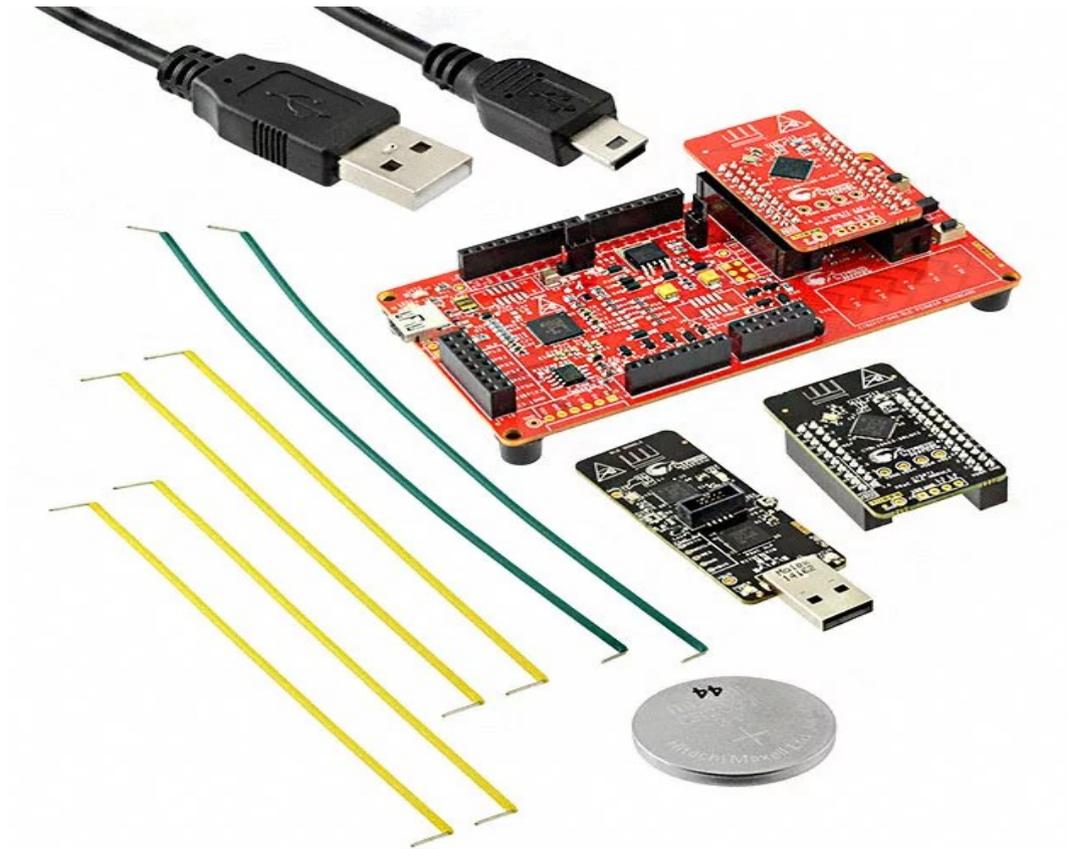
Bluetooth Hardware



- * Several Solutions available, including your phone!
- * Cypress Semiconductor, Nordic Semiconductor, TI, Silicon Labs, etc. all have Bluetooth Radios
- * Almost all of them have Eddystone Reference Designs that are freely available

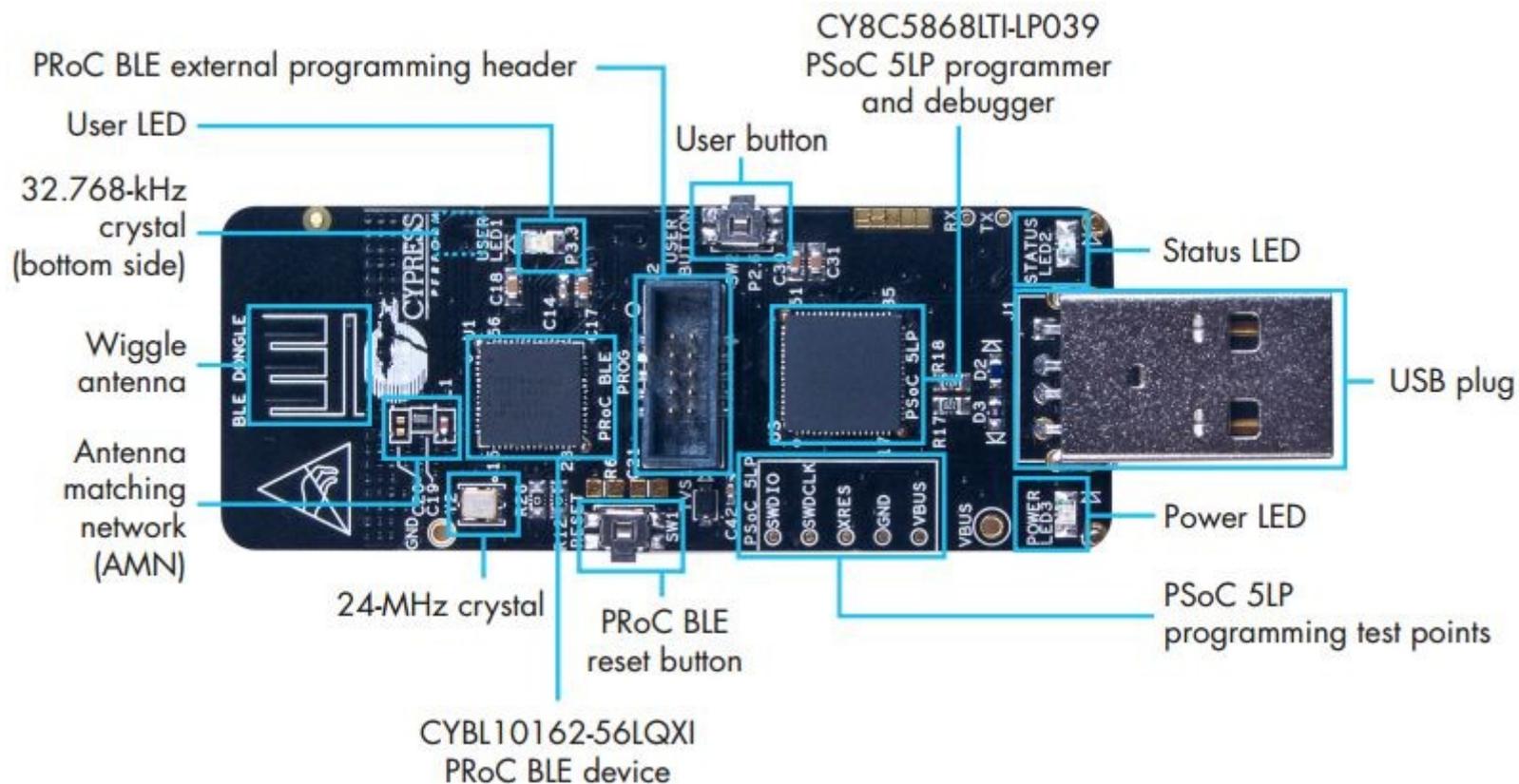
Cypress CY8CKIT-042-BLE

- * Cypress Programmable System-on-Chip
- * Malleable hardware that you can program!



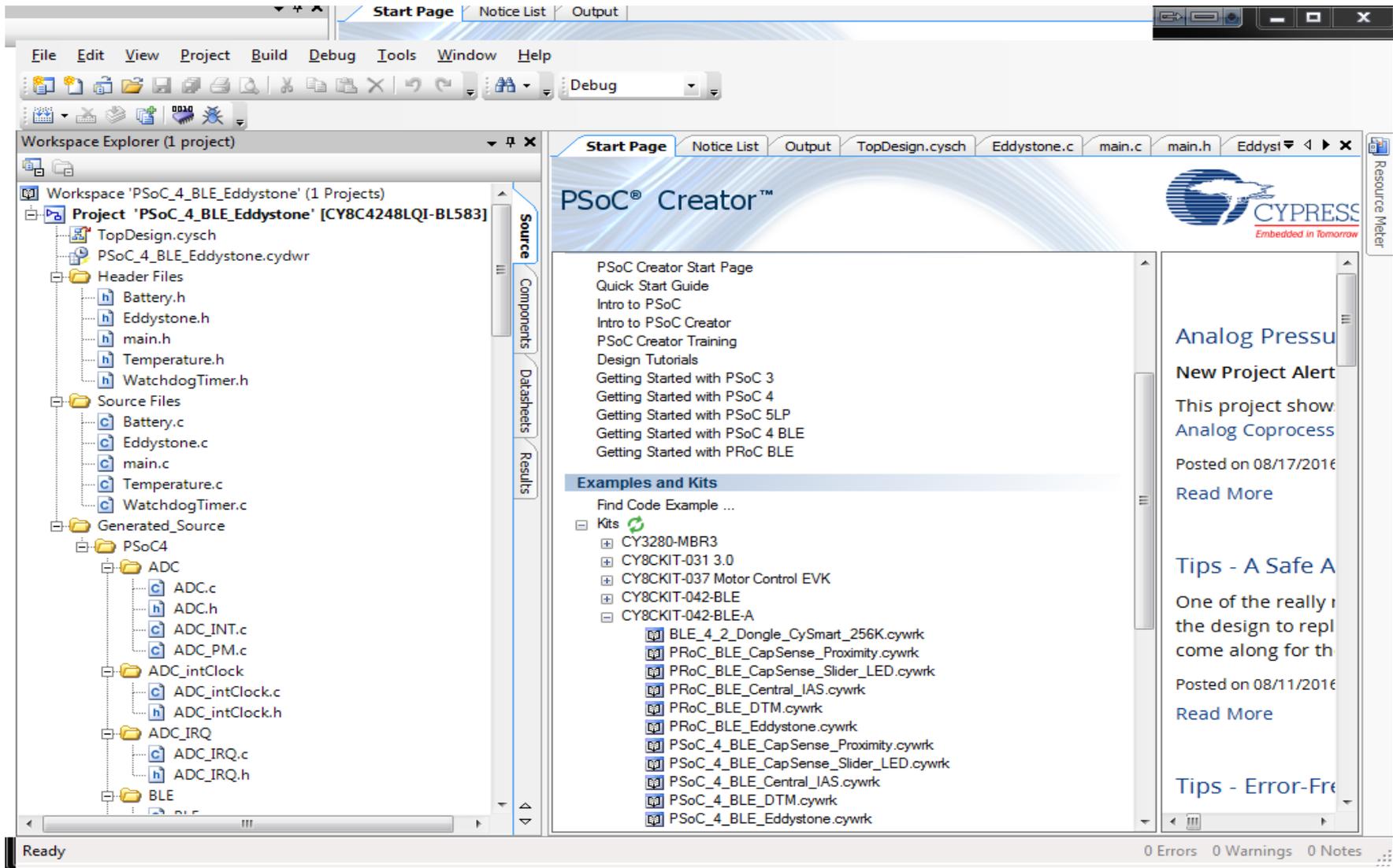
Cypress Smart Dongle

- * Cypress Programmable System-on-Chip
- * Built-in JTAG Programmer



Source: Cypress

PSoC Creator



Source: Cypress

Creator Schematic

The screenshot displays the Cypress PSoC Creator IDE interface. The main workspace shows a schematic page titled "Page 1" with the following components and descriptions:

- BLE:** BLE Component is configured as a Beacon in non-connectable advertisement mode with no timeout (continuous transmission). The component is represented by a "Bluetooth 4.2" icon.
- ADC Interface:** ADC measures battery voltage and die temperature. The diagram shows the ADC SAR Seq component connected to the VREF pin of the ADC and the DieTemp component.
- Watch Dog Timer:** Watchdog timer is used for waking up the system when the system is in low power mode. The component is represented by a "WDT" icon with a "Global Signal" and "WDT_interrupt" output.
- LEDs:** The red LED indicates that device is advertising connectable Adv packets. The green LED indicates that device is advertising URL / UID packets. The blue LED indicates that device is advertising TLM packets. The diagram shows three LEDs (RED, GREEN, BLUE) connected to the VDD pin through resistors.
- Button:** SW2 to change the Adv type to connectable. The component is represented by a "SW2" icon with a "ConnectionAdv_ISR" output.

The left sidebar shows the Project Explorer with the following structure:

- Workspace PSoC_4_BLE_Eddystone (1 Projects)
 - Project 'PSoC_4_BLE_Eddystone' (CYBC4)
 - TopDesign.cysch
 - PSoC_4_BLE_Eddystone.cydw
 - Header Files
 - Battery.h
 - Eddystone.h
 - main.h
 - Temperature.h
 - WatchdogTimer.h
 - Source Files
 - Battery.c
 - Eddystone.c
 - main.c
 - Temperature.c
 - WatchdogTimer.c
 - Generated_Source
 - PSoC4
 - ADC
 - ADC.c
 - ADC.h
 - ADC_INT.c
 - ADC_PM.c
 - ADC_intClock
 - ADC_intClock.c
 - ADC_intClock.h
 - ADC_IRQ
 - ADC_IRQ.c
 - ADC_IRQ.h
 - BLE
 - BLE.c
 - BLE.h
 - BLE_custom.c
 - BLE_custom.h
 - BLE_eventHandler.c
 - BLE_eventHandler.h

Source: Cypress

BLE Component Design

Configure 'BLE'

Name: BLE

General Profiles **GAP Settings** L2CAP Settings Advanced Built-in

Advertisement data settings:

Name	Value
<input checked="" type="checkbox"/> Flags	
<input checked="" type="checkbox"/> General discoverable mode	
<input checked="" type="checkbox"/> BR/EDR not supported	
<input type="checkbox"/> Local Name	
<input type="checkbox"/> TX Power Level	
<input type="checkbox"/> Slave Connection Interval Range	
<input checked="" type="checkbox"/> Service UUID	
<input checked="" type="checkbox"/> Eddystone	
<input type="checkbox"/> Eddystone Configuration	
<input type="checkbox"/> Service Solicitation	
<input checked="" type="checkbox"/> Service Data	
<input checked="" type="checkbox"/> Eddystone	
Data	10:F2:00:70:61:74:74:65:72:6E:61:67:65:6E:74:73:00
<input type="checkbox"/> Eddystone Configuration	
<input type="checkbox"/> Service Manager TK Value	
<input type="checkbox"/> Appearance	
<input type="checkbox"/> Public Target Address	
<input type="checkbox"/> Random Target Address	
<input type="checkbox"/> Advertising Interval	
<input type="checkbox"/> LE Bluetooth Device Address	

Advertisement packet:

Description	Value	Index
AD Data 1: <<Flags>>		
Length	0x02	[0]
<<Flags>>	0x01	[1]
BR/EDR not supported General discoverable mode	0x06	[2]
AD Data 2: <<Complete list of 16-bit UUIDs available>>		
Length	0x03	[3]
<<Complete list of 16-bit UUIDs available>>	0x03	[4]
Service: Eddystone		
[0]	0xAA	[5]
[1]	0xFE	[6]
AD Data 3: <<Service Data>>		
Length	0x14	[7]
<<Service Data>>	0x16	[8]
Service: Eddystone		
[0]	0xAA	[9]
[1]	0xFE	[10]
Data: 10:F2:00:70:61:74:74:65:72:6E:61:67:65:6E:74:73:00		
[0]	0x10	[11]
[1]	0xF2	[12]
[2]	0x00	[13]

Restore Defaults

Datasheet

OK Apply Cancel

Page 1

Ready 0 Errors 0 Warnings 0 Notes

Source: Cypress

GAP/GATT Configuration

The screenshot displays the PSoC Creator 3.3 interface. The main window shows the 'Configure BLE' dialog box, which is used for configuring Bluetooth Low Energy (BLE) services. The dialog is titled 'Configure BLE' and has a 'Name' field set to 'BLE'. It features several tabs: 'General', 'Profiles', 'GAP Settings', 'L2CAP Settings', 'Advanced', and 'Built-in'. The 'Profiles' tab is currently selected, showing a tree view of the BLE profile structure. The tree includes a 'Server' profile with various characteristics such as 'Generic Access', 'Device Name', 'Appearance', 'Peripheral Preferred Connection Parameters', 'Generic Attribute', 'Service Changed', 'Client Characteristic Configuration', 'Eddystone', 'Eddystone Configuration', 'Lock State', 'Lock', 'Unlock', 'URI Data', 'URI Flags', 'Advised Tx Power Levels', 'Tx Power Mode', 'Beacon Period', and 'Reset'. The 'Service' field is set to 'Custom Service' with a UUID of 'FEAA' and a 16-bit length. The 'Service type' is set to 'Primary'. The 'Included services' list contains 'Eddystone Configuration'. A 'Datasheet' button is located at the bottom left of the dialog. The background shows the PSoC Creator workspace with a project tree on the left and a component catalog on the right. The component catalog is open to the 'Interface' section, showing a schematic diagram of an ADC SAR Seq component. The diagram includes a 12-bit SAR component connected to a 16-bit ADC. The component is labeled 'ADC SAR Seq' and has pins for 'soc', 'sdone', and 'eoc'. The schematic also shows a 12-bit SAR component connected to a 16-bit ADC. The 'LEDs' section of the component catalog shows a schematic diagram of three LEDs (RED, GREEN, BLUE) connected to a Vdd supply through resistors (2.2K, 1.5K, 1.5K). The status bar at the bottom indicates '0 Errors, 0 Warnings, 0 Notes'.

Source: Cypress

Test the Beacon Firmware

The screenshot shows the Beacon Manager app interface. At the top, there is a dark blue header with a hamburger menu icon on the left, the text "Beacon Manager" in the center, and a vertical ellipsis icon on the right. Below the header, the main content area is divided into two sections, each representing a beacon. Each section starts with a Wi-Fi signal icon and a signal strength indicator (e.g., -84 or -82). The first section is for "Eddystone URL" and lists the URL, battery level, number of packets, temperature, and active time. The second section is for "PatternAgent" and lists the Eddystone UID, namespace ID, instance ID, URL, battery level, number of packets, temperature, and active time.

Beacon Manager

Eddystone URL
URL: <https://bit.ly/2b4edZk>

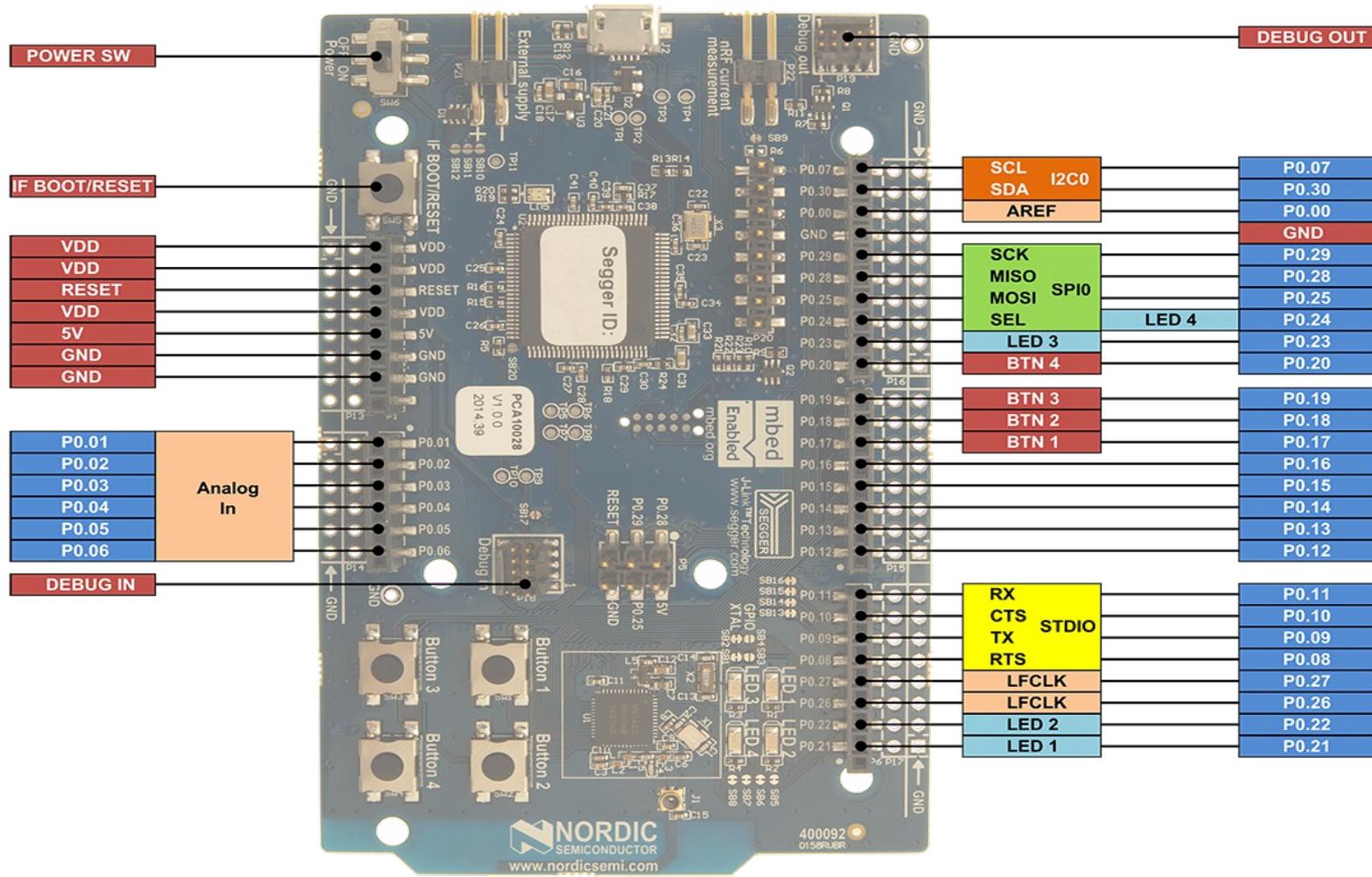
Eddystone TLM
Battery: 3377 mV
Packets: 130880
Temperature: 25.0
Active since: 2016-09-08 13:11:04

PatternAgent
Eddystone UID
Namespace ID: c45bbe114575ffae5489
Instance ID: 000000000001

Eddystone URL
URL: <https://bit.ly/2b4edZk>

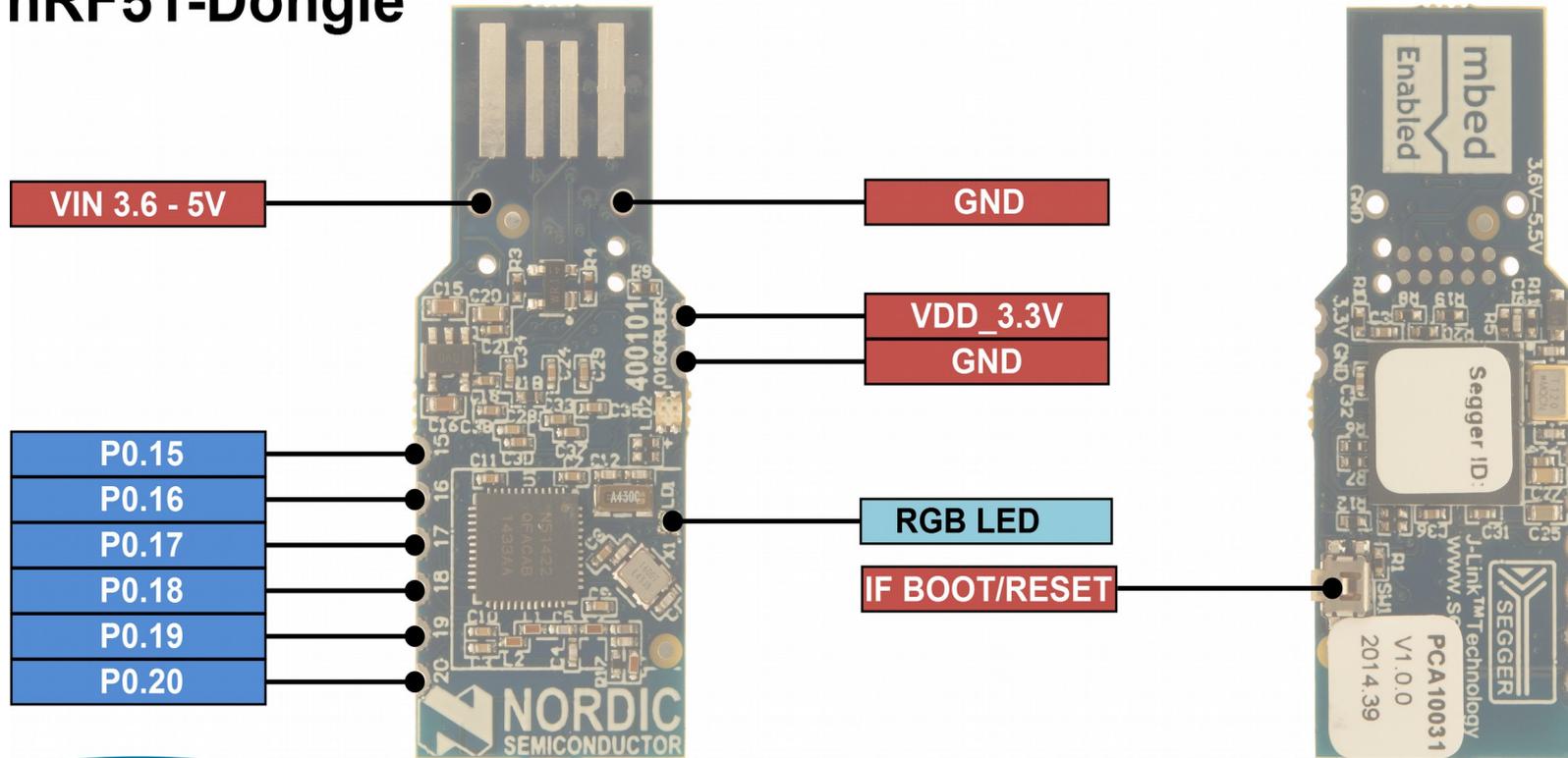
Eddystone TLM
Battery: 0 mV
Packets: 5841
Temperature: -127.0
Active since: 2016-09-09 09:30:29

Nordic nRF51 Series



Nordic nRF51 Series

nRF51-Dongle



ARM mBed Online

- * We'll use the ARM mbed on-line tools
- * Nothing to install, just create account
- * We'll import the Google Reference code
- * The boards enumerate via USB as Mass Storage device class
- * Just download the compiled hex file to the Mass Storage class device (i.e. E:) and it programs the firmware for you, Easy!

ARM mbed Example

The screenshot shows the ARM mbed website interface. The browser address bar displays the URL: <https://developer.mbed.org/teams/mbed-os-examples/code/mbed-os-example-ble-EddystoneService/>. The page header includes the ARM mbed logo and navigation links for Hardware, Documentation, Code, Questions, Forum, and Compiler. The main content area features a user profile for 'mbed-os-examples' and the repository name 'mbed-os-example-ble-EddystoneService'. A description states: 'Eddystone beacons broadcast a small amount of information, like URLs, to nearby BLE devices. The canonical source for this example lives at https://github.com/ARMmbed/mbed-os-example-ble/tree/master/BLE_EddystoneService'. Below this is a navigation menu with 'Home', 'History', 'Graph', 'API Documentation', 'Wiki', and 'Pull Requests'. The main text explains that the application runs in two stages: 1. On startup, the Configuration Service (which allows modification of the beacon) runs for a user-defined period (default: 30 seconds). 2. When the Configuration Service period ends, the Eddystone Service broadcasts advertisement packets. A section titled 'Running the application' includes 'Requirements' (installing the 'Physical Web' application on a phone) and 'Building instructions' (hardware requirements in the main readme). On the right side, there is a 'Repository toolbox' with buttons for 'Import into Compiler', 'Export to desktop IDE', 'Build repository', '+ Follow', and 'Embed url: `https://program.*/teams/mbed-os`'. Below that is a 'Repository details' table.

Repository details	
Type:	Program
Created:	26 Jul 2016
Imports:	16
Forks:	0
Commits:	10
Dependents:	0
Dependencies:	0

Source: ARM

Select your Board

The screenshot shows the mbed IDE interface. The main window displays the 'Select a Platform' dialog box. The dialog is titled 'Select a Platform' and shows the 'Nordic nRF51-DK' platform selected. The dialog includes a description of the kit, a list of registered platforms, and a 'Select Platform' button. The background shows the IDE interface with a file explorer on the left and a program workspace in the center.

Select a Platform

Nordic nRF51-DK

You are currently compiling for the Nordic nRF51-DK platform.

Description

The nRF51 Development Kit is a single-board development kit for Bluetooth Smart, ANT and 2.4GHz proprietary applications using the nRF51 Series SoC. This kit supports both development for both nRF51822 and nRF51822 SoCs.

The kit is compatible with the Arduino Uno Rev3 on 3 standard, making it possible to use 3rd party shields that are compatible to this standard with the kit.

The kit supports the standard Nordic Software Development Tool-chain using IAR, IAR and GCC. The kit also supports ARM mbed tool-chain for rapid prototyping and development using mbed's cloud-based IDE and tool-chain with an extensive range of open-source software libraries. Program/Debug options on the kit are Segger J-Link Lite for standard tool-chain and CMSIS-DAP for mbed. The kit gives access to all I/O and interfaces via connectors and has 4 LEDs and 4 buttons which are user-programmable.

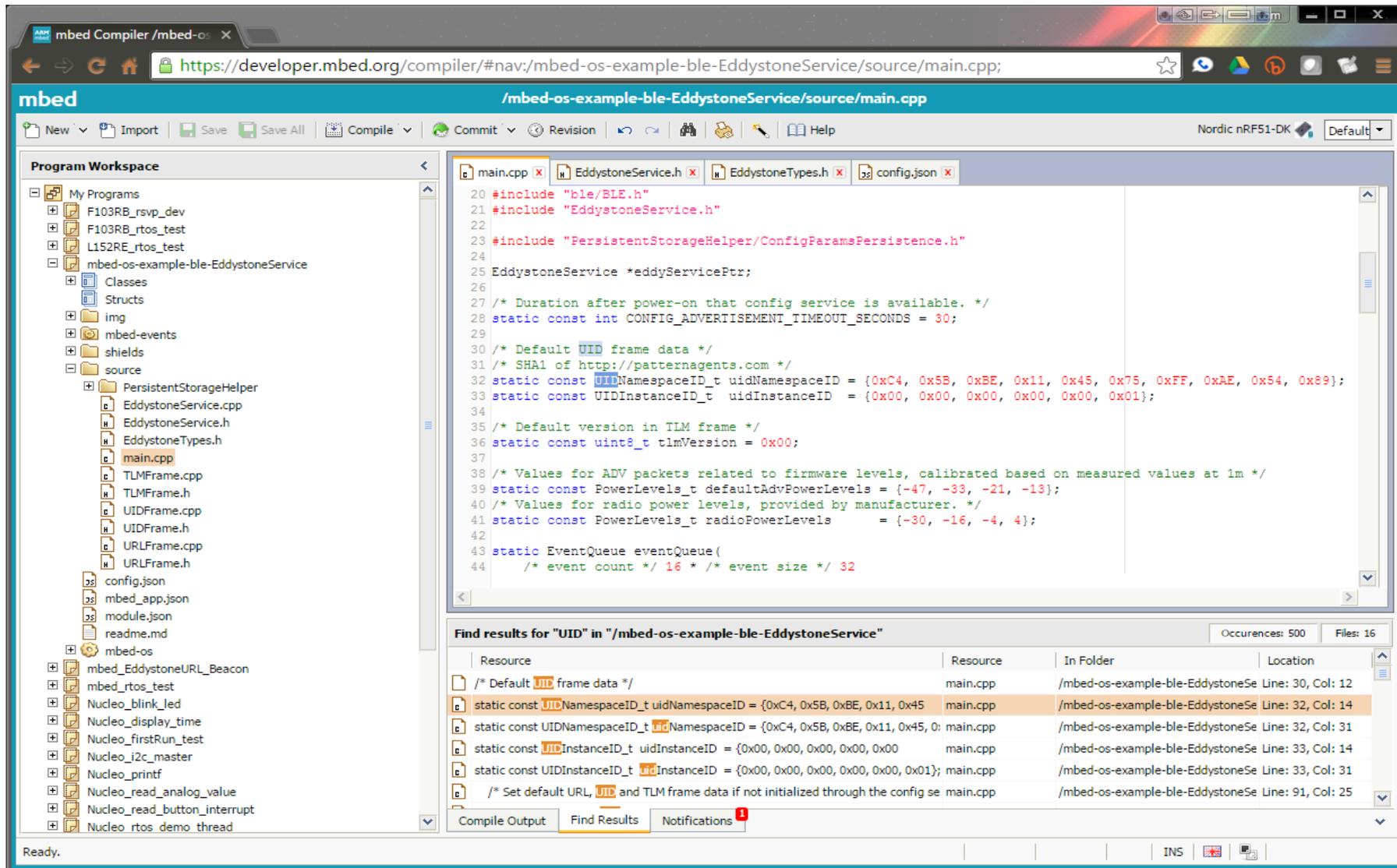
A range of software examples are available from the nRF51-DK to support Bluetooth Smart, ANT and 2.4GHz applications.

nRF51-DK

Your registered platforms

Platform	Status
u-blox C027	Available
NUCLEO-F103RB	Available
NUCLEO-L152RE	Available
NUCLEO-L492RE	Available
NUCLEO-F401RE	Available
NUCLEO-F411RE	Available
Nordic nRF51-DK	Selected
Nordic nRF51-	Available
Add Platform	Available

Change UUID



The screenshot shows the mbed IDE interface. The main window displays the source code for `main.cpp` in the `/mbed-os-example-ble-EddystoneService/source` directory. The code includes headers for BLE and EddystoneService, and defines a default UUID. A search results window is open at the bottom, showing occurrences of "UID" in the code.

```
20 #include "ble/BLE.h"
21 #include "EddystoneService.h"
22
23 #include "PersistentStorageHelper/ConfigParamsPersistence.h"
24
25 EddystoneService *eddyServicePtr;
26
27 /* Duration after power-on that config service is available. */
28 static const int CONFIG_ADVERTISEMENT_TIMEOUT_SECONDS = 30;
29
30 /* Default UID frame data */
31 /* SHA1 of http://patternagents.com */
32 static const UIDNamespaceID_t uidNamespaceID = {0xC4, 0x5B, 0xBE, 0x11, 0x45, 0x75, 0xFF, 0xAE, 0x54, 0x89};
33 static const UIDInstanceID_t uidInstanceID = {0x00, 0x00, 0x00, 0x00, 0x00, 0x01};
34
35 /* Default version in TLM frame */
36 static const uint8_t tlmVersion = 0x00;
37
38 /* Values for ADV packets related to firmware levels, calibrated based on measured values at 1m */
39 static const PowerLevels_t defaultAdvPowerLevels = {-47, -33, -21, -13};
40 /* Values for radio power levels, provided by manufacturer. */
41 static const PowerLevels_t radioPowerLevels = {-30, -16, -4, 4};
42
43 static EventQueue eventQueue(
44     /* event count */ 16 * /* event size */ 32
```

Find results for "UID" in "/mbed-os-example-ble-EddystoneService"

Resource	Resource	In Folder	Location
/* Default UID frame data */	main.cpp	/mbed-os-example-ble-EddystoneSe	Line: 30, Col: 12
static const UIDNamespaceID_t uidNamespaceID = {0xC4, 0x5B, 0xBE, 0x11, 0x45, 0x75, 0xFF, 0xAE, 0x54, 0x89};	main.cpp	/mbed-os-example-ble-EddystoneSe	Line: 32, Col: 14
static const UIDNamespaceID_t uidNamespaceID = {0xC4, 0x5B, 0xBE, 0x11, 0x45, 0x75, 0xFF, 0xAE, 0x54, 0x89};	main.cpp	/mbed-os-example-ble-EddystoneSe	Line: 32, Col: 31
static const UIDInstanceID_t uidInstanceID = {0x00, 0x00, 0x00, 0x00, 0x00, 0x01};	main.cpp	/mbed-os-example-ble-EddystoneSe	Line: 33, Col: 14
static const UIDInstanceID_t uidInstanceID = {0x00, 0x00, 0x00, 0x00, 0x00, 0x01};	main.cpp	/mbed-os-example-ble-EddystoneSe	Line: 33, Col: 31
/* Set default URL, UID and TLM frame data if not initialized through the config se	main.cpp	/mbed-os-example-ble-EddystoneSe	Line: 91, Col: 25

Change URL

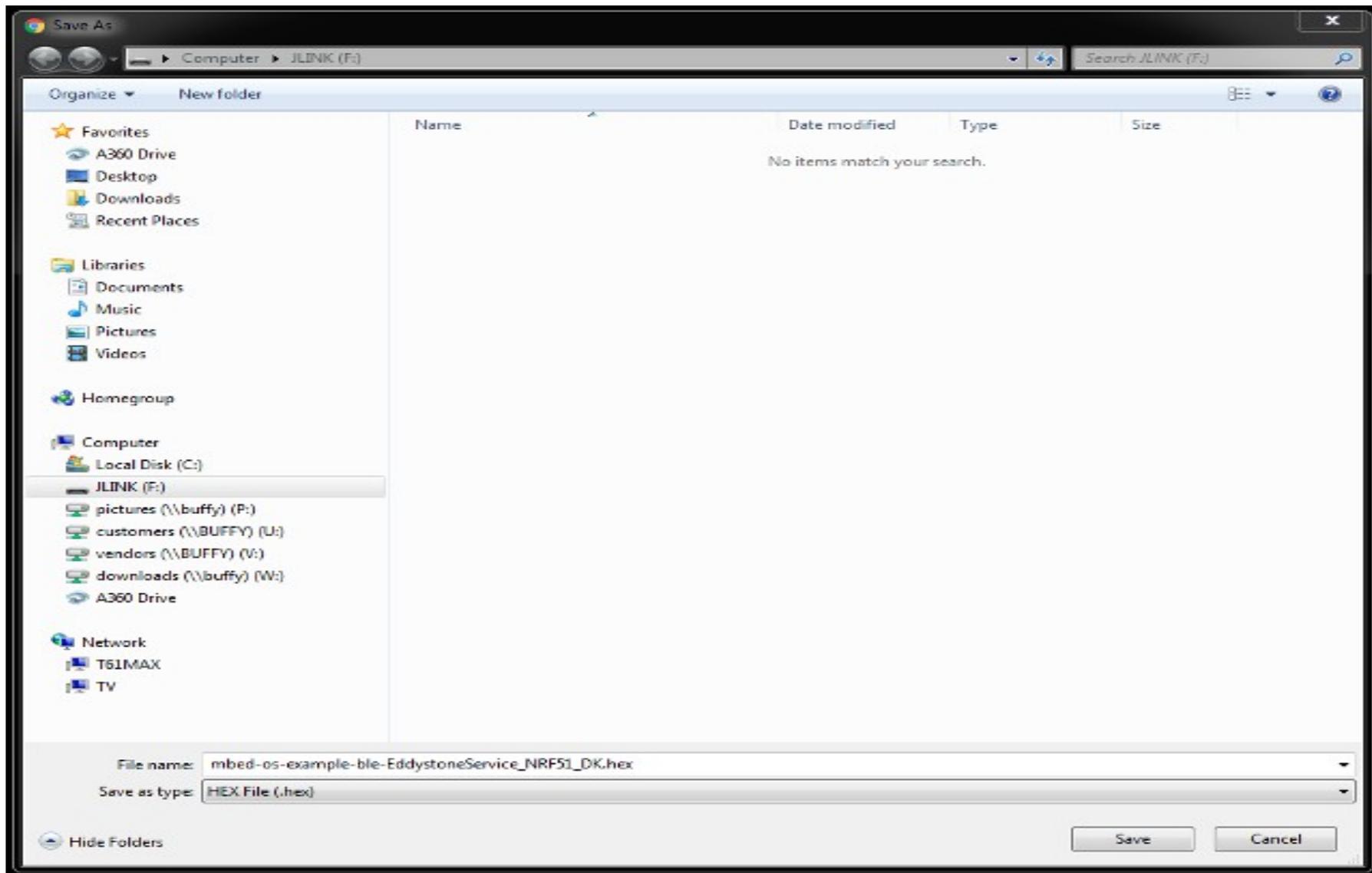
The screenshot shows the mbed Compiler IDE interface. The main window displays the source file `EddystoneTypes.h` with the following code:

```
16
17 #ifndef __EDDYSTONE_TYPES_H__
18 #define __EDDYSTONE_TYPES_H__
19
20 #include <stdint.h>
21 #include <stddef.h>
22
23 #ifndef YOTTA_CFG_EDDYSTONE_DEFAULT_DEVICE_NAME
24     #define YOTTA_CFG_EDDYSTONE_DEFAULT_DEVICE_NAME "PatternAgent"
25 #endif
26
27 #ifndef YOTTA_CFG_EDDYSTONE_DEFAULT_URL
28     #define YOTTA_CFG_EDDYSTONE_DEFAULT_URL "https://bit.ly/2b4edZk"
29 #endif
30
31 /**
32  * Macro to expand a 16-bit Eddystone UUID to 128-bit UUID.
33  */
34 #define UUID_URL_BEACON(FIRST, SECOND) {
35     0xee, 0x0c, FIRST, SECOND, 0x87, 0x86, 0x40, 0xba,
36     0xab, 0x96, 0x99, 0xb9, 0x1a, 0xc9, 0x81, 0xd8,
37 }
38
39 /**
40  * Eddystone 16-bit UUID.
```

The search results pane at the bottom shows the following table:

Resource	Resource	In Folder	Location
<code>const char* url = YOTTA_CFG_EDDYSTONE_DEFAULT_URL;</code>	<code>main.cpp</code>	<code>/mbed-os-example-ble-EddystoneSe</code>	Line: 92, Col: 33
<code>#ifndef YOTTA_CFG_EDDYSTONE_DEFAULT_URL</code>	<code>EddystoneTypes.h</code>	<code>/mbed-os-example-ble-EddystoneSe</code>	Line: 27, Col: 19
<code>#define YOTTA_CFG_EDDYSTONE_DEFAULT_URL "https://bit.ly/2b4edZk"</code>	<code>EddystoneTypes.h</code>	<code>/mbed-os-example-ble-EddystoneSe</code>	Line: 28, Col: 23
<code>const char DEFAULT_URL[] = YOTTA_CFG_EDDYSTONE_DEFAULT_URL;</code>	<code>EddystoneTypes.h</code>	<code>/mbed-os-example-ble-EddystoneSe</code>	Line: 102, Col: 38
<code>#ifndef YOTTA_CFG_EDDYSTONE_DEFAULT_URL_FRAME_INTERVAL</code>	<code>EddystoneService.h</code>	<code>/mbed-os-example-ble-EddystoneSe</code>	Line: 35, Col: 19
<code>#define YOTTA_CFG_EDDYSTONE_DEFAULT_URL_FRAME_INTERVAL 700</code>	<code>EddystoneService.h</code>	<code>/mbed-os-example-ble-EddystoneSe</code>	Line: 36, Col: 23

Download and Run



Test the Beacon Firmware

The screenshot shows the Beacon Manager app interface. At the top, there is a dark blue header with a hamburger menu icon on the left, the text "Beacon Manager" in the center, and a vertical ellipsis icon on the right. Below the header, the main content area is divided into two sections, each representing a beacon. Each section starts with a Wi-Fi signal strength icon and a signal strength value (-84 and -82 respectively). The first section is for "Eddystone URL" and lists the URL, battery level, number of packets, temperature, and active time. The second section is for "PatternAgent" and lists the Eddystone UID, namespace ID, instance ID, URL, battery level, number of packets, temperature, and active time.

Beacon Manager

Eddystone URL
URL: <https://bit.ly/2b4edZk>

Eddystone TLM
Battery: 3377 mV
Packets: 130880
Temperature: 25.0
Active since: 2016-09-08 13:11:04

PatternAgent
Eddystone UID
Namespace ID: c45bbe114575ffae5489
Instance ID: 000000000001

Eddystone URL
URL: <https://bit.ly/2b4edZk>

Eddystone TLM
Battery: 0 mV
Packets: 5841
Temperature: -127.0
Active since: 2016-09-09 09:30:29

Creekside Five Beehive

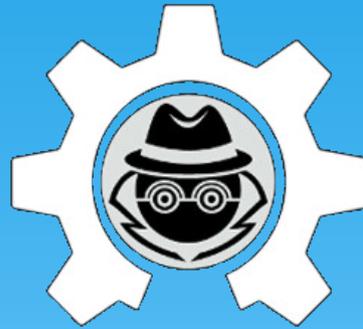


- * The Beacon is set to describe the Beehive just outside
- * It's a simple page, just a demo
- * We'll be adding metrics to it as time allows...

Summary

- * Beacons are just starting to take off
- * IMHO, Eddystone will win in 2-3 years time
- * Decentralized, open standard
- * Low barriers to entry
- * Proximity Beacon API is awesome, well designed
- * Attachments provide more data to apps than the beacon is capable of in hardware, a nice idea!
- * Attachments can be JSON packets or whatever...
- * “One beacon, Many apps!”
- * It's easy, go ahead and deploy your own beacons!

Questions?



* Thank You!

Questions

- * Are World Bee Populations actually increasing?

Personally, I don't think there is enough data on wild hives to make that assertion; however, (perhaps in response to increased CCD awareness) the number of new, “managed” hives introduced in the US is currently exceeding the CCD die-off rate of older and existing “managed” hives.

- * Many more people are introducing backyard and urban hives in the U.S. and elsewhere; and are volunteering to share data on their hives, so – more hives...

(AgPro & FAOSTAT data - <http://faostat3.fao.org/>)

Questions ?

- * Are RF emissions safe for Bees and/or People?

Clearly, exposure to any *high power* RF source can be a problem for living things, but are the 2.4Ghz signals of Bluetooth and Wi-Fi radios high enough power to be a problem for Bees? In my experience, there has been no discernible difference between adjacent hives, one in close proximity to the transmitter, and the other 6-8 feet away.

(Hardly a study... just anecdotal...)

However, the transmitter works best “outside” the hive, so an external antennae placement is preferred.

Beacons for the Bees

For PDXMakerWeek, we will be talking about working with the new Google EddyStone Beacon technology and other physical web applications, like Beacons for Beehives. Google Beacons create a simple way to "tag" the physical location of objects, such as wild and domestic Beehives. We'll demonstrate ways to monitor and Geo-tag Beehives.

To participate in the presentation, you can:

- bring a Bluetooth Enabled Smart Phone
- install the latest version of Google Chrome
- enable Physical Web Notifications in Google Chrome (see: support.google.com/chrome/answer/6239299)

RSVP REQUIRED:

Wednesday, September 14, 2016

6:00 PM to 8:00 PM



@PatternAgents

Open Source Design Patterns

