

Opportunities for Libraries in the Internet of Things

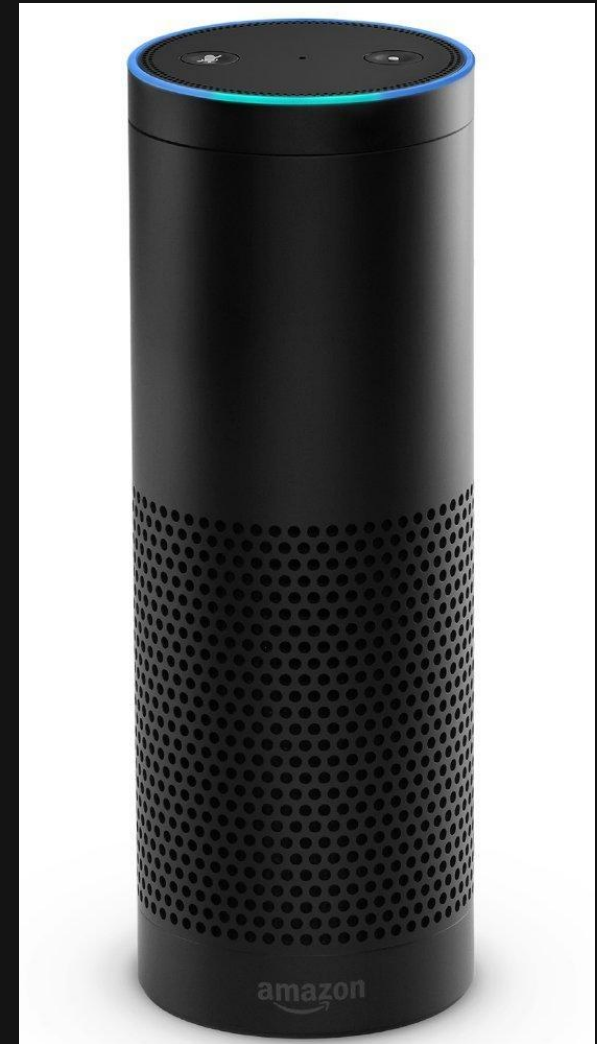
ARL Hunchery 2016

Adam Rogers
*Emerging Technology
Services Librarian*

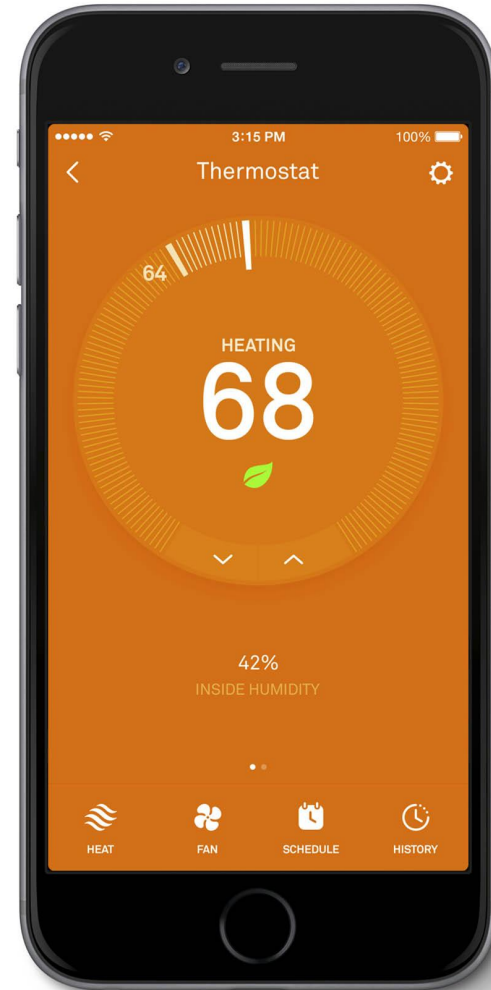
 **NCSU LIBRARIES**
go.ncsu.edu/make

What is the Internet of Things?

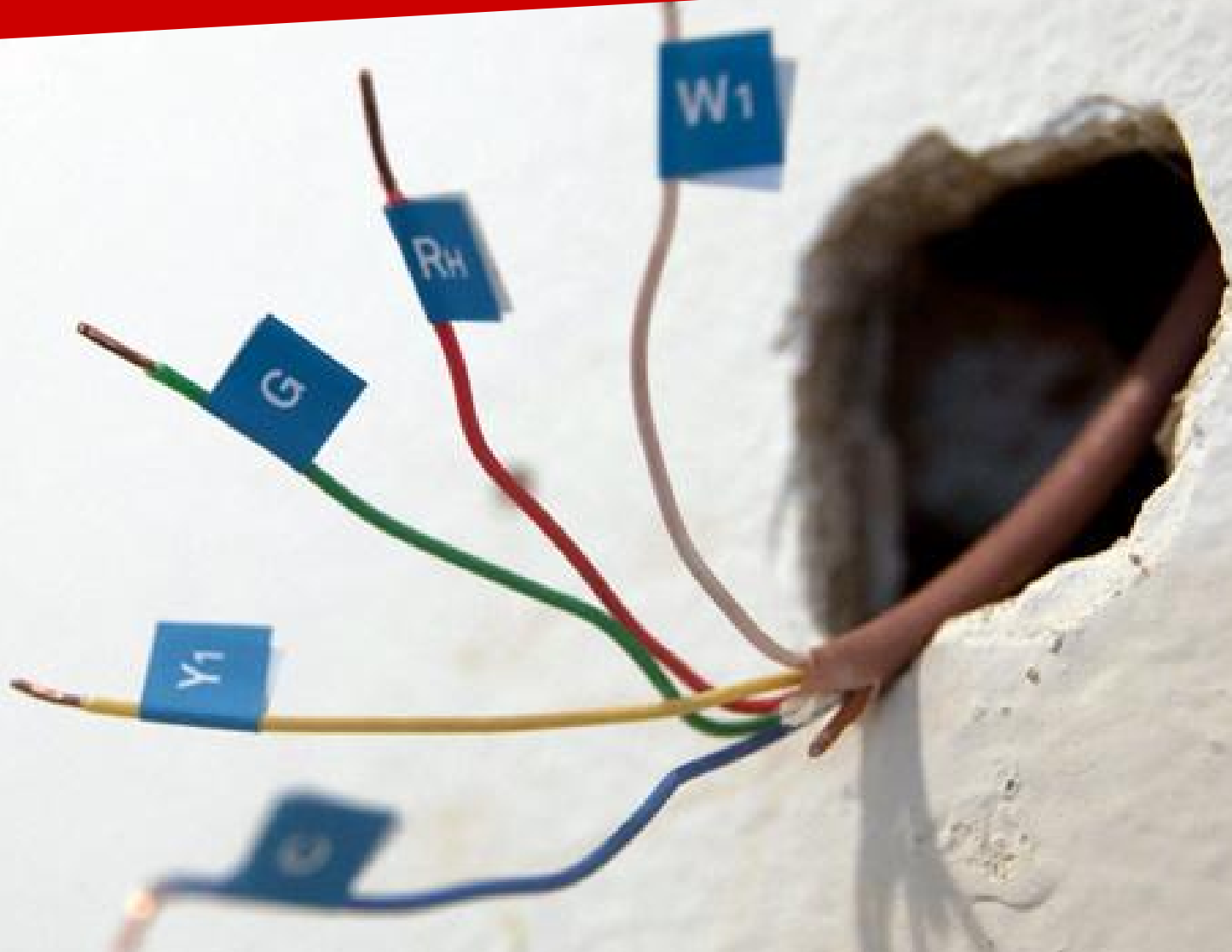
- “smart” physical things
- informed by web data
- sensing and reporting to the web



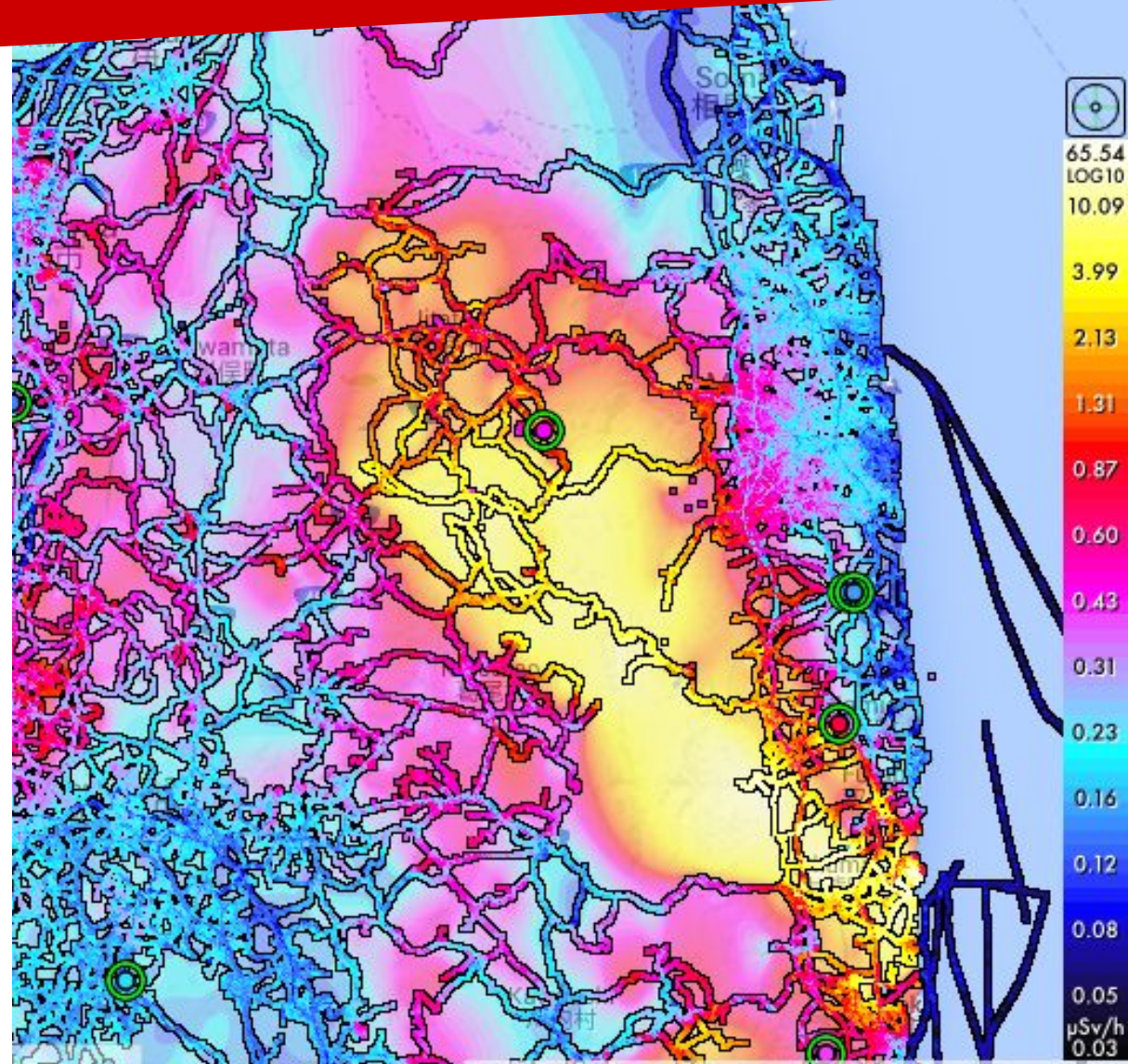
Example Thing: Nest



Example Thing: Nest



Example Thing: Safecast



Example Thing: Safecast

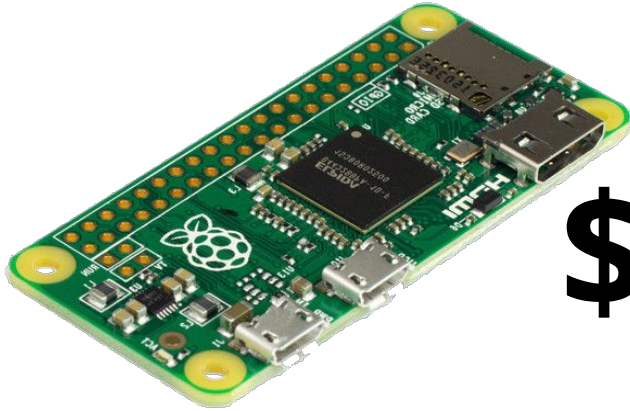
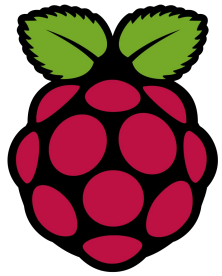


IoT Basics

- **Hardware:** tiny integrated computer
- **Sensors:** radiation, temperature, motion, humidity, microphones, etc.
- **Actuators:** lights, sounds, relays, etc.
- **Connection:** WiFi, Bluetooth, cellular
- **Data/Web Platform:** API, website, mobile app, dashboard

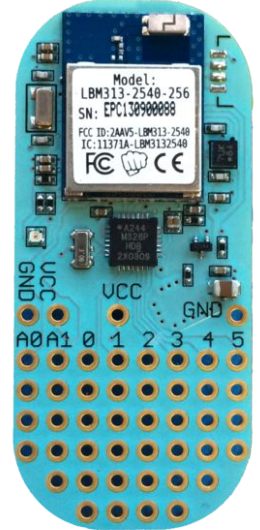
Why should you care?

It's Radically Affordable



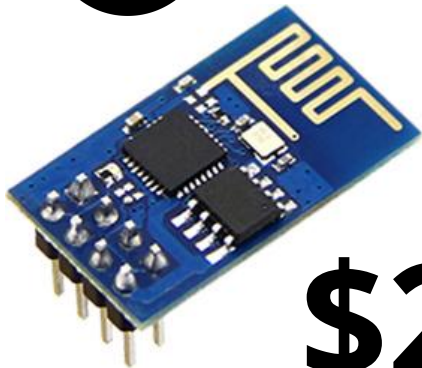
\$5

\$30



open source
hardware

\$70



\$2



The Coding is Easy*

```
ESP8266_Simple_Button_Light | Arduino 1.6.8

ESP8266_Simple_Button_Light

// constants won't change. They're used here to set pin numbers:
const int buttonPin = 10; // the number of the pushbutton pin; note pin 10 here is pin SD3
const int ledPin = 2; // the number of the LED pin; note pin 2 here is pin D4 on ESP8266

// variables will change:
int buttonState = 0; // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  pinMode(ledPin, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
}

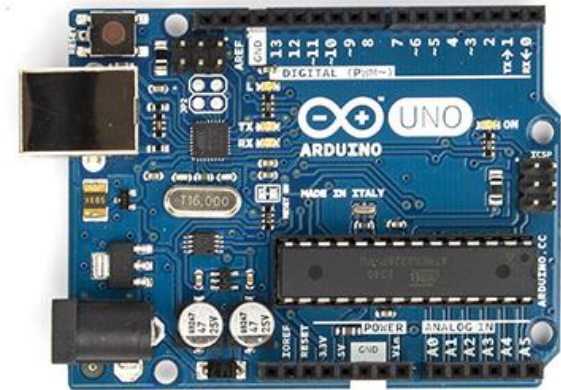
void loop() {
  // read the state of the pushbutton value:
  buttonState = digitalRead(buttonPin);

  // check if the pushbutton is pressed.
  // if it is, the buttonState is HIGH:
  if (buttonState == HIGH) {
    // turn LED on:
    digitalWrite(ledPin, HIGH);
    delay(1000); // this will leave the light on for 1 second after the button is pressed
  } else {
    // turn LED off:
    digitalWrite(ledPin, LOW);
  }
}

Done uploading.

at java.net.AbstractPlainDatagramSocketImpl.join(AbstractPlainDatagramSocketImpl.java:
at java.net.MulticastSocket.joinGroup(MulticastSocket.java:323)
at javax.jmdns.impl.JmDNSImpl.openMulticastSocket(JmDNSImpl.java:463)
at javax.jmdns.impl.JmDNSImpl.<init>(JmDNSImpl.java:420)

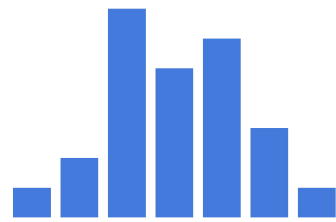
NodeMCU 1.0 (ESP-12E Module), 80 MHz, 115200, 4M (3M SPIFFS) on /dev/cu.wchusbserial1410
```



* esp. for web developers & digital libraries folks

Rich Data Platforms

IFTTT

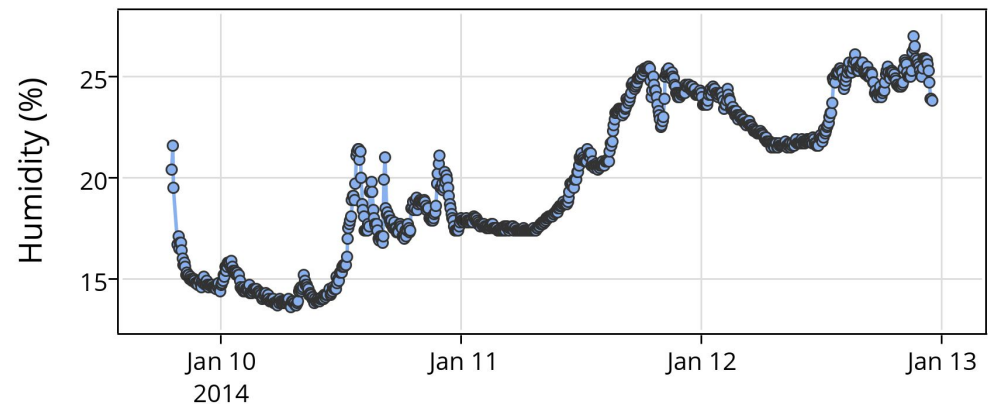
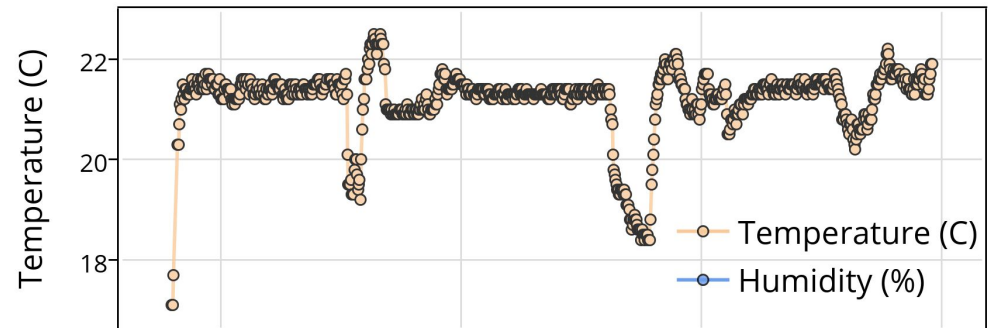


plotly



ThingSpeak

Plotly Arduino API
Temperature and Humidity



Library Possibilities

- Solving Problems
- Collecting Assessment Data
- New Service Models
- In-Building Navigation
- Expertise to Serve our Communities

Example Projects

Card Access System



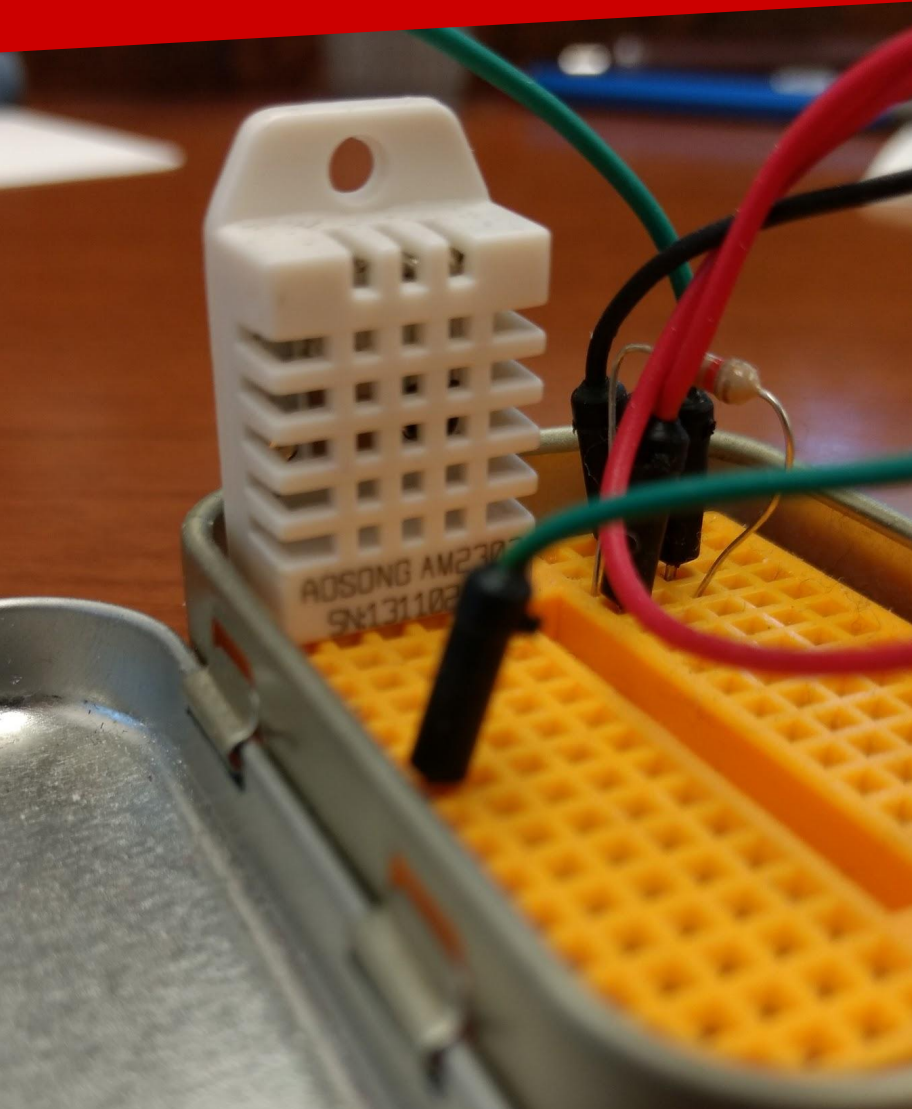
- Mediates access to restricted space
- Generates great assessment data
- Student-built; ~\$100 in parts
- Open Source! see github.com/NCSU-Libraries

Door Counter



- Motion sensor counting visitors
- Great data on space use, traffic flows
- Student-built, ~\$100 in parts
- Open Source! see github.com/NCSU-Libraries

Environmental Monitor



- Reports temperature & humidity data from offsite storage
- Sends warnings at specific thresholds
- Staff-built, <\$75
- *BETA*—still being refined and tested

ARL Possibilities

- Sharing Projects & Open Source Code
- Training Librarians
- Standardized Data Collection
- A Shared Data Platform?



THANKS!
**I look forward to
your comments!**



Slides: go.ncsu.edu/ARL_IOT

Me: adam_rogers@ncsu.edu

Web: go.ncsu.edu/make