



THE CUSTOMER SUCCESS PLATFORM
SALES SERVICE MARKETING COMMUNITY ANALYTICS APPS

Build Salesforce IoT applications with ARM mbed™

Build a Salesforce IoT application with ARM mbed in less than an hour!

Doug Anson

IoT Architect – ARM

doug.anson@arm.com

Sam Grove

Senior Applications Engineer – ARM

sam.grove@arm.com

dreamforce®

Objective

- * Introduction to the MBED online developer environment
- * Initial setup of device and developer environment
- * Compile and connect your device to the Salesforce cloud
- * Further reading and fun...

Checklist

- ✓ Browser: Chrome Browser, Firefox, Safari, Internet Explorer...
- ✓ Windows Users: Please install this USB driver onto your box
 - ✓ http://developer.mbed.org/media/downloads/drivers/mbedWinSerial_16466.exe
- ✓ Putty Serial Terminal – installed and ready...

✓ Ethernet cable at the ready...

✓ A workshop kit:



- 3 RFID Tags
- 1 USB Cable
- 1 K64F with RFID reader shield
- Container

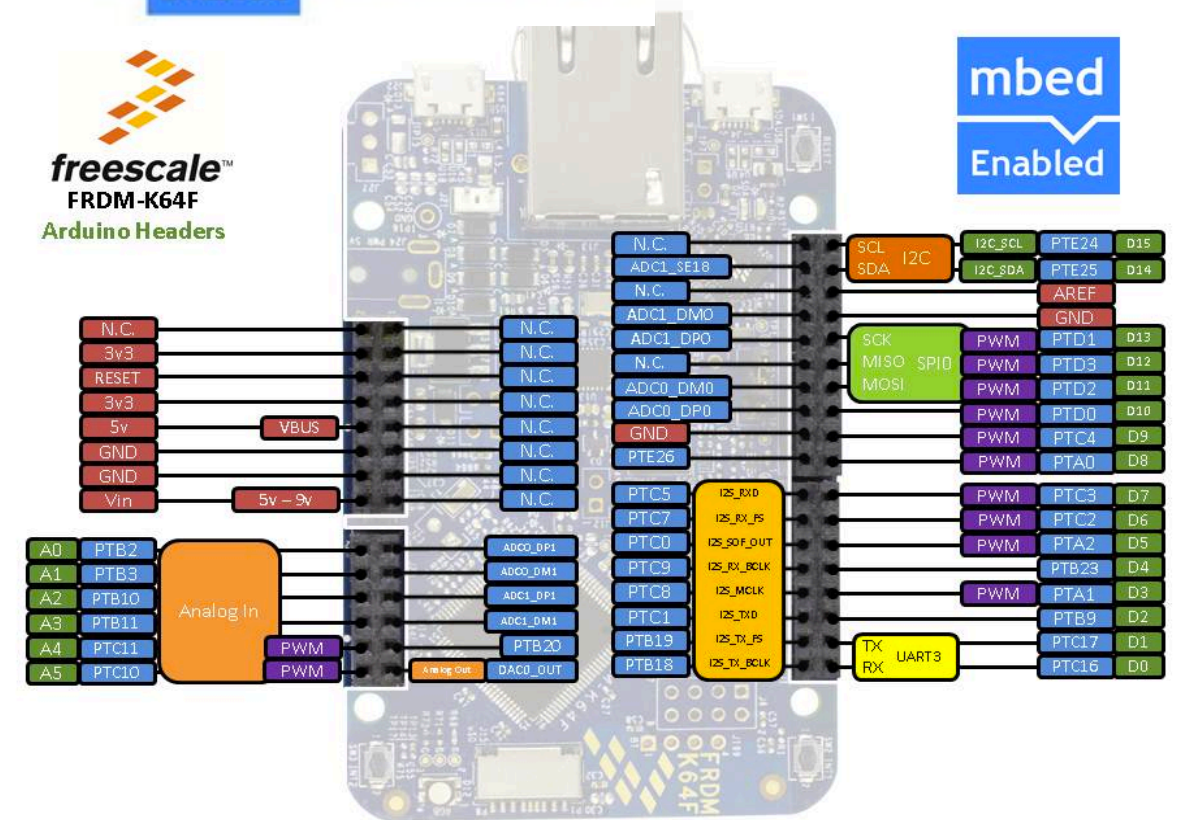
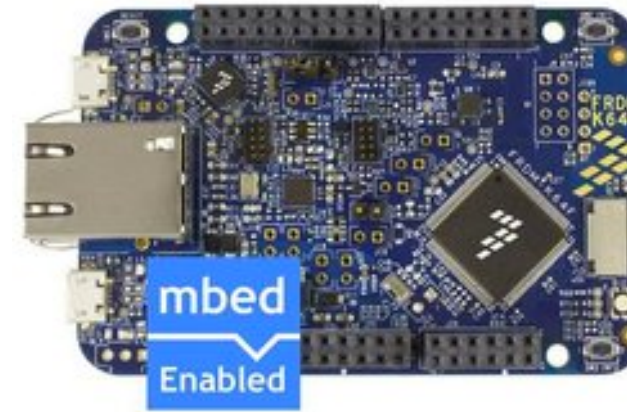
mbed Online Developer Environment



- * Cloud-based
- * Supports multiple compile targets (platforms)
- * Full source code control
- * Searchable source repository
- * Components
- * Online forum, wiki

The image shows two overlapping browser windows. The top window displays the ARM mbed Developer Site, which includes a navigation menu (Platforms, Components, Handbook, Cookbook, Code, Questions, Forum), a search bar, and a user profile (Hi, ansond). The main content area features a 'Getting Started' button and a section titled 'ARM mbed Developer Site' with a description of the platform and a link to 'Announcing our plans for mbed v3.0'. A 'Blog' section is also visible.

The bottom window shows the mbed Compiler IDE. The 'Program Workspace' on the left lists a project structure with folders like 'EndpointMain' and files like 'main.cpp'. The main editor displays the source code for 'main.cpp', which includes headers for 'HTTPClient' and 'StatusReporter', and a 'mainTask' function. The 'Compile Output' window at the bottom shows a table with columns for 'Description', 'Error Number', 'Resource', 'In Folder', and 'Location'. The status bar at the bottom indicates 'Ready' and 'In 1 col 1 92 INS'.



mbed Device – What is it?

- * 120 MHz 32bit ARM Cortex M4
- * 256KB RAM, 1MB Flash
- * Ethernet
- * MicroSD card reader
- * CAN, USB, UART, I2C, SPI
- * Analog Input & Output
- * Multi-color LED
- * PWM output
- * Enumerates as USB mass storage
- * R3 shield compatible

Oh... One more thing...

- * ARM would like to give each workshop attendee a FRDM-K64F! Just:
- * Complete the workshop...
- * ...return your workshop “kit” intact (thanks!!)
- * And we’ll give you your own fresh/boxed K64F



Enjoy and Have Fun!

ARM[®] mbed[™]

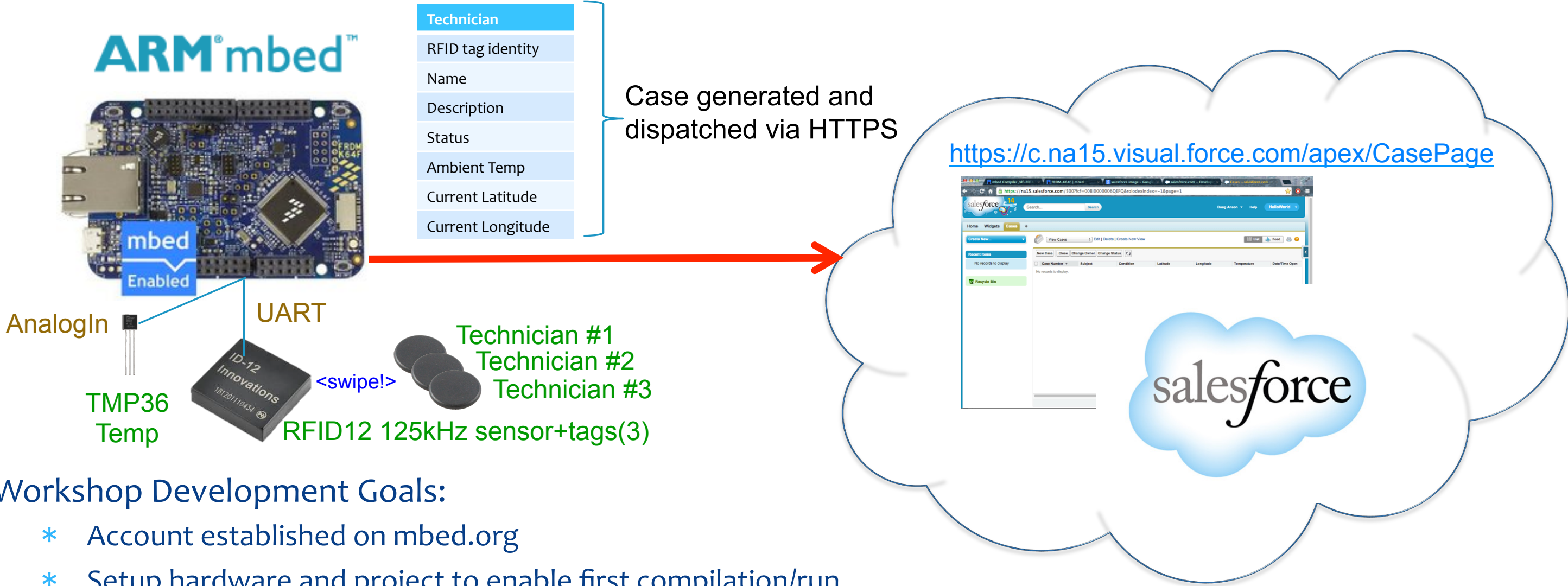
Workshop Scenario: RFID-based Case Generator

ARMmbed™

- * Technicians utilize IoT-connected RFID-based intelligent street lights to convey technician sign-in and service status/condition to the service company through Salesforce.com Cases.
- * Utilize RFID tags and the K64F (as part of the street light design) in a service dispatch scenario where you will use “sign in” into the light as a technician to indicate servicing of the next street light in your service roster. Each street light in your design is equipped to hold 3 unique technician’s RFID identities.
- * The K64F+RFID reader will scan your RFID tag and send an specific Case dispatch indicating a technician’s ID and will include the light’s current ambient temperature (in Celsius) in addition to the technicians’ onsite status and description.



Workshop RFID-based Case Generator...



Workshop Development Goals:

- * Account established on mbed.org
- * Setup hardware and project to enable first compilation/run
- * Update the sample program for your 3 RFID identifiers– each will represent a uniquely identified technician
- * Confirm by swiping RFID tags to generate various Salesforce.com Cases
- * Demonstrate sensor integration by including the ambient temperature as part of the Case content

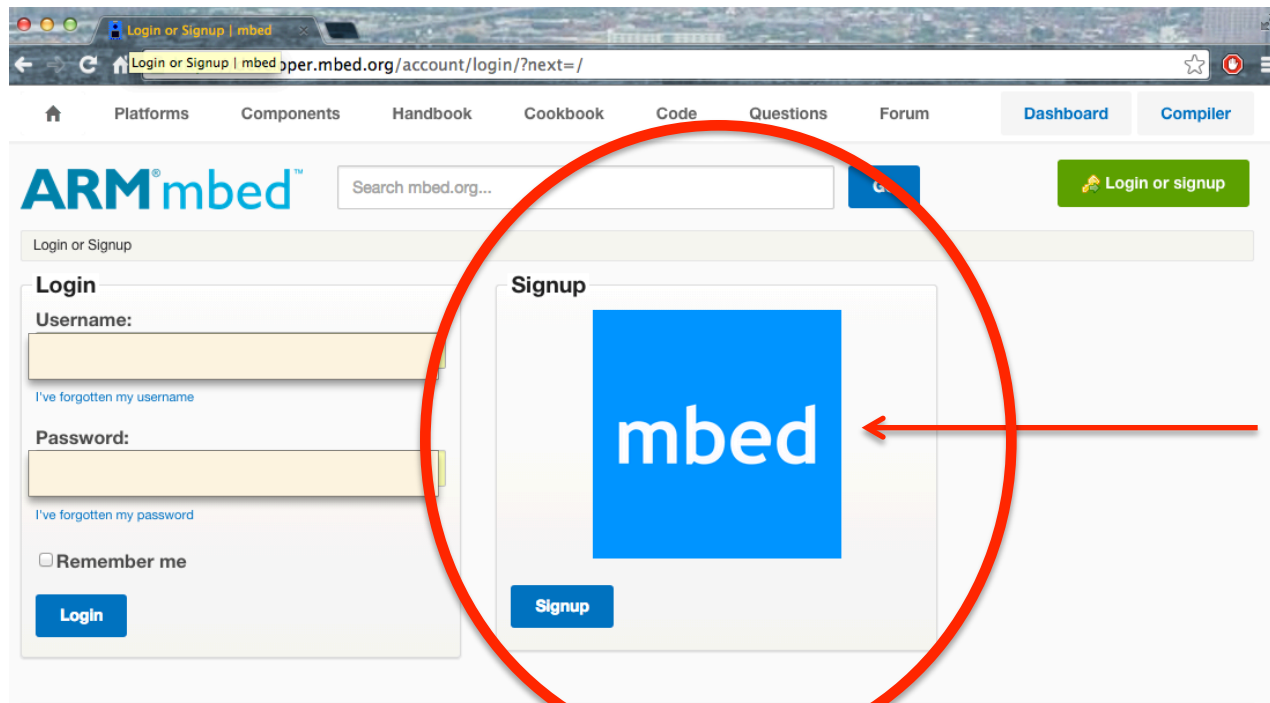


Workshop Hardware Setup

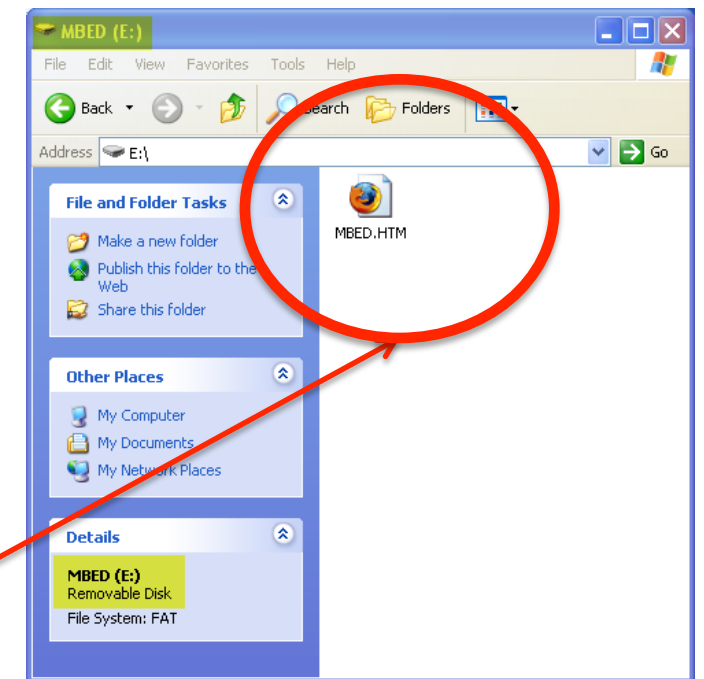
USB cable goes HERE



MBED Account Setup/Login



Choose this and create your free developer account

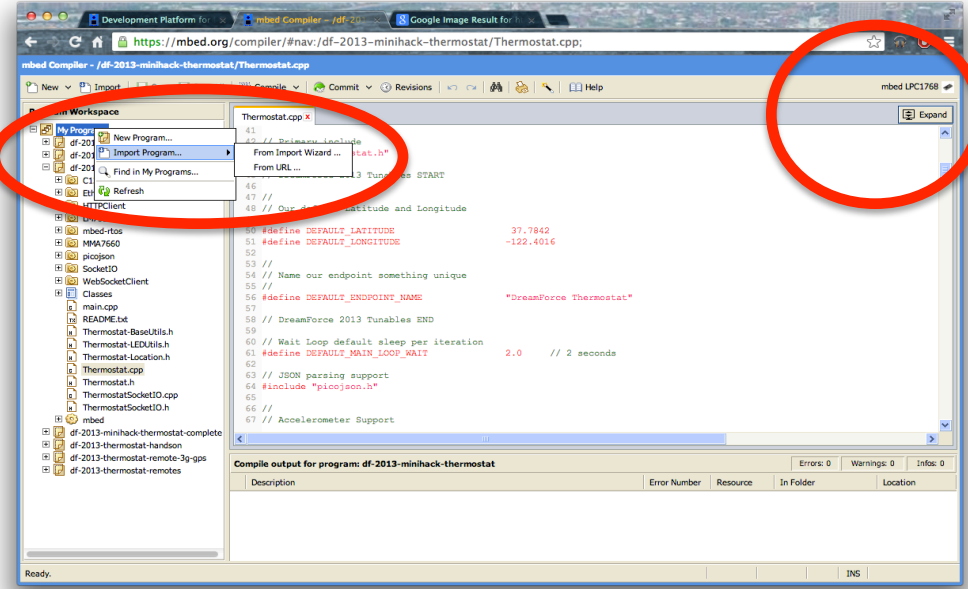


View your MBED mounted drive
- MBED.htm: device information

ARM mbed™

Workshop Code

Right-Click "My Programs"
Select: "Import from Wizard"



"FRDM K64F" compile target

Select first row Press
"Import from URL"

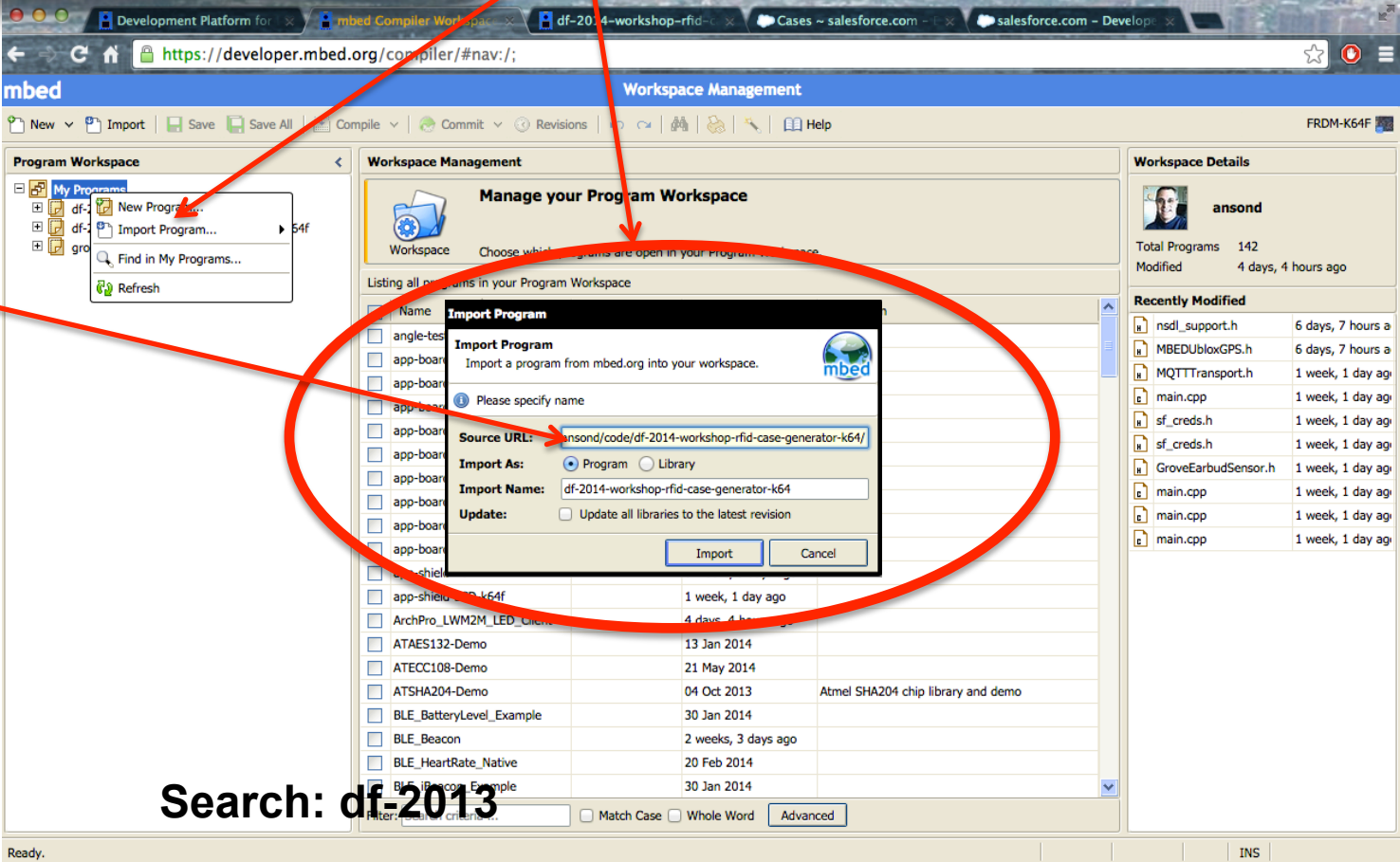
* Import Project (via URL):

http://developer.mbed.org/teams/MBED_DEMOS/code/df-2014-rfid-case-gen-k64f-exercise/

* Import, then select, and Compile->Compile All

* Copy binary to mbed USB drive (note behavior...)

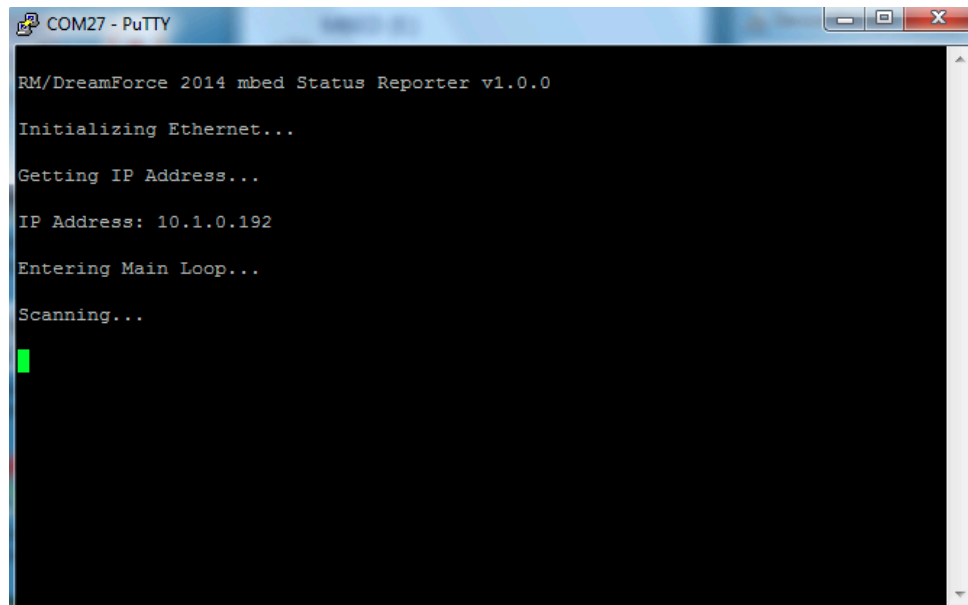
* Reset button press to run... then...



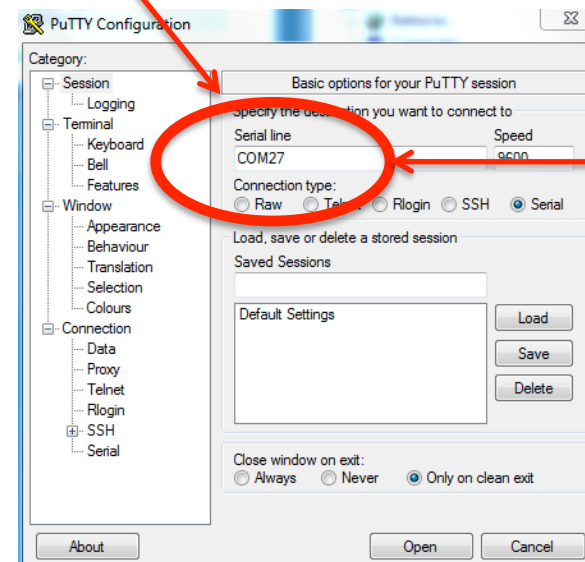
Search: df-2013

Workshop Running Code

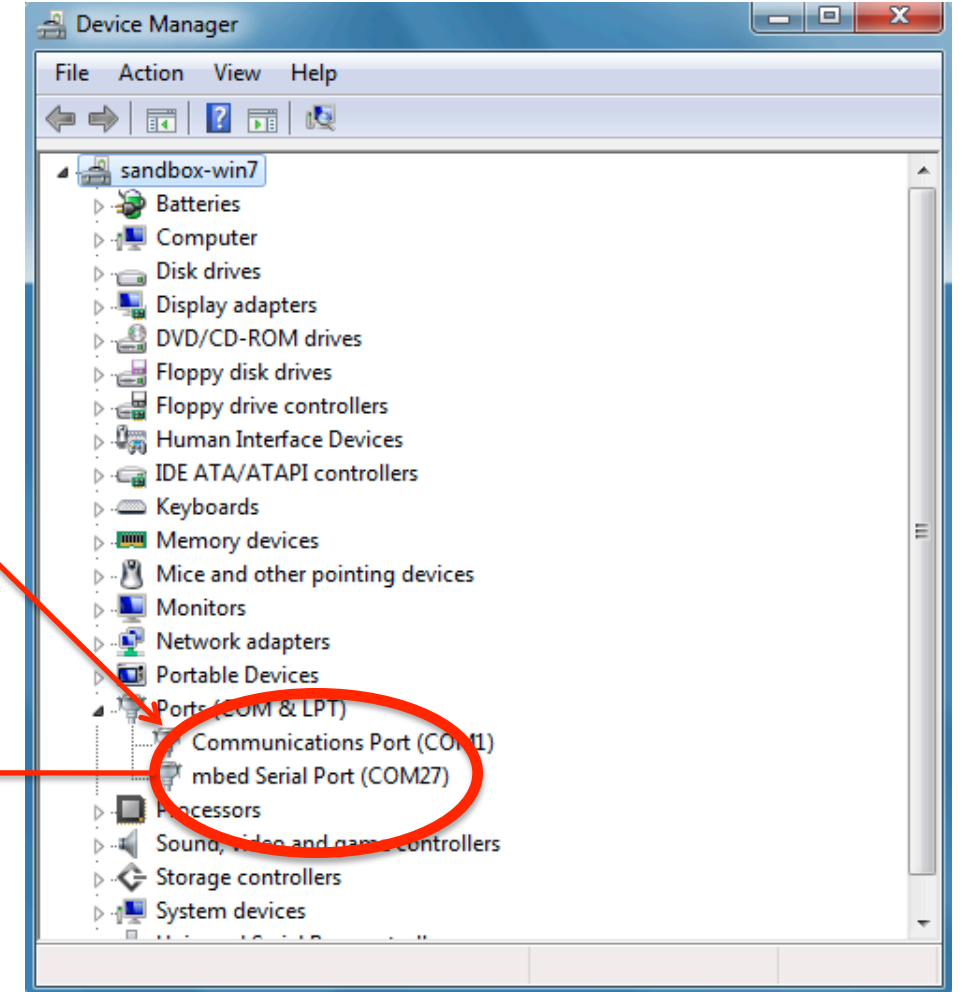
- * Windows Device Manager: Ports
- * Find “mbed Serial Port”, note: COM^{XY}
- * Launch Putty: connect to COM^{XY}
- * All else defaulted: 9600/8/1
- * Reset button again. You should see something like this:



Putty Console



Putty Configuration Panel

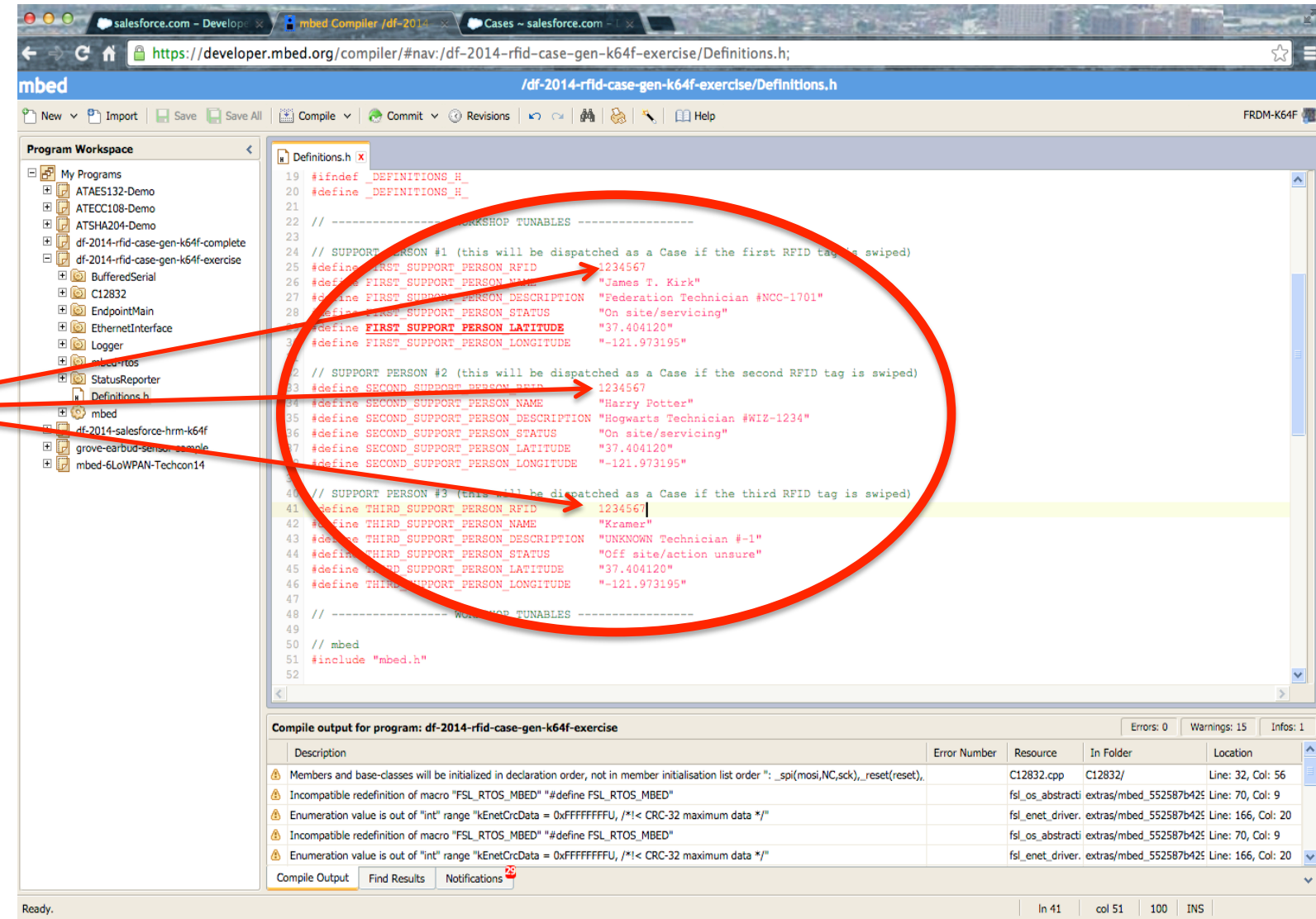


Windows Device Manager

A Green LED indicates that your K64F is initialized and on the network!

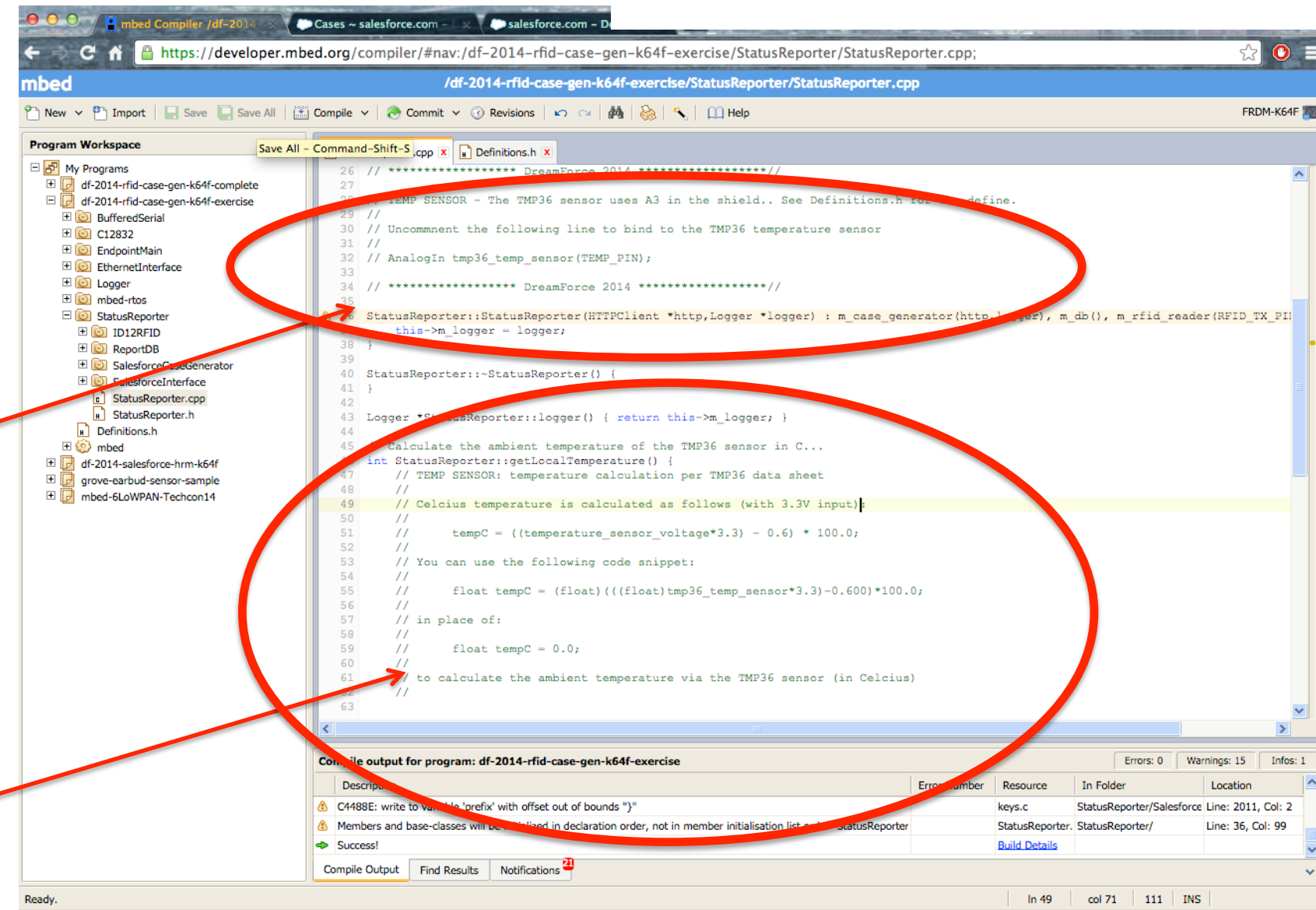
Workshop Modification #1

- * Edit Definitions.h by double-clicking
- * Swipe each RFID tag and view the Putty serial console
- * Record the RFID ID value for each tag
- * Update each of the 3 supported technician identities listed. Additionally, you can customize each technician's status and description if desired.



Workshop Modification #2

- * We want to read the temperature sensor and send the temperature (in Celsius) as part of the Case
- * Uncomment the AnalogIn definition to access the temperature sensor
- * Complete the “getLocalTemperature()” function per comments



Now, watch your Cases being created in Salesforce.com (Presentation Viewer)

<https://c.na15.visual.force.com/apex/CasePage>

Congrats! You are Done!

Takeaways

- * The mbed online developer environment is modern, powerful, easy to use and **COOL**.
- * mbed devices are very easy to develop with... and very little setup is required.
- * mbed devices can talk directly to the Salesforce.com cloud using REST and HTTPS.
 - * Anonymous Case generation through RFID token presentation used for our example.

Next Steps – use this project as a template to build your own IoT applications!

- * mbed Salesforce API: http://developer.mbed.org/teams/MBED_DEMOS/code/SalesforceInterface/
- * Geo-located HeartRate Viewer Example using above API (See it run in the ARM booth)
 - * http://developer.mbed.org/teams/MBED_DEMOS/code/df-2014-salesforce-hrm-k64f/

Thank you!

(doug.anson@arm.com, sam.grove@arm.com)

 Thank You

ARM mbed™