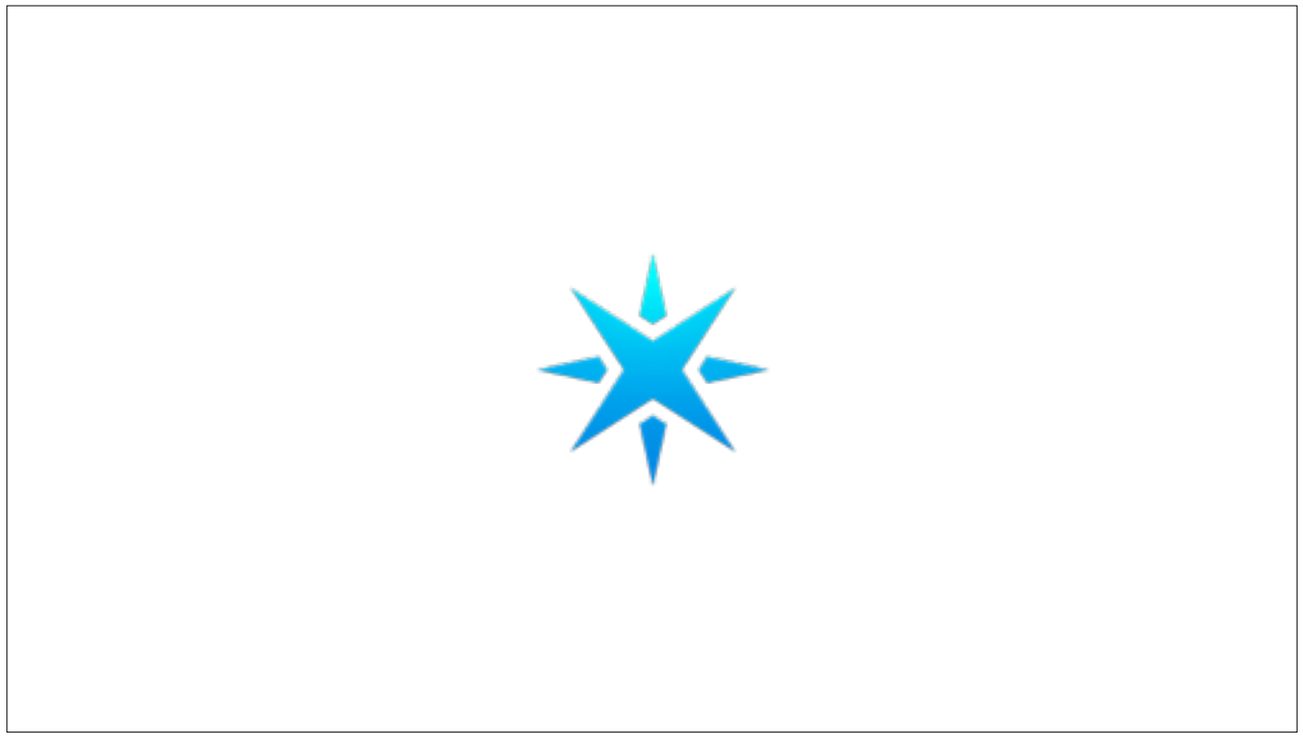


[Spark Core]

SPARKCORE WORKSHOP

Fabian Morón Zirfas
University of Applied Sciences Potsdam (Germany)
2014

Hi I’m Fabian Morón Zirfas

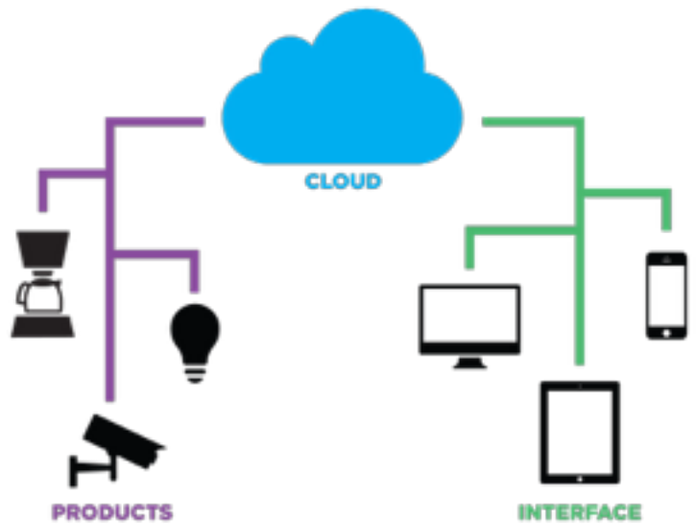
TOPICS

1. What is a Spark Core?
2. How to get it?
3. Demo
4. Prerequisites
5. Command Line Interface
6. Setup, Claim & Blink
7. Walk through code examples (firmware & web)
8. Trix & Hints
9. Ups & Downs &ToDos
10. Hands On

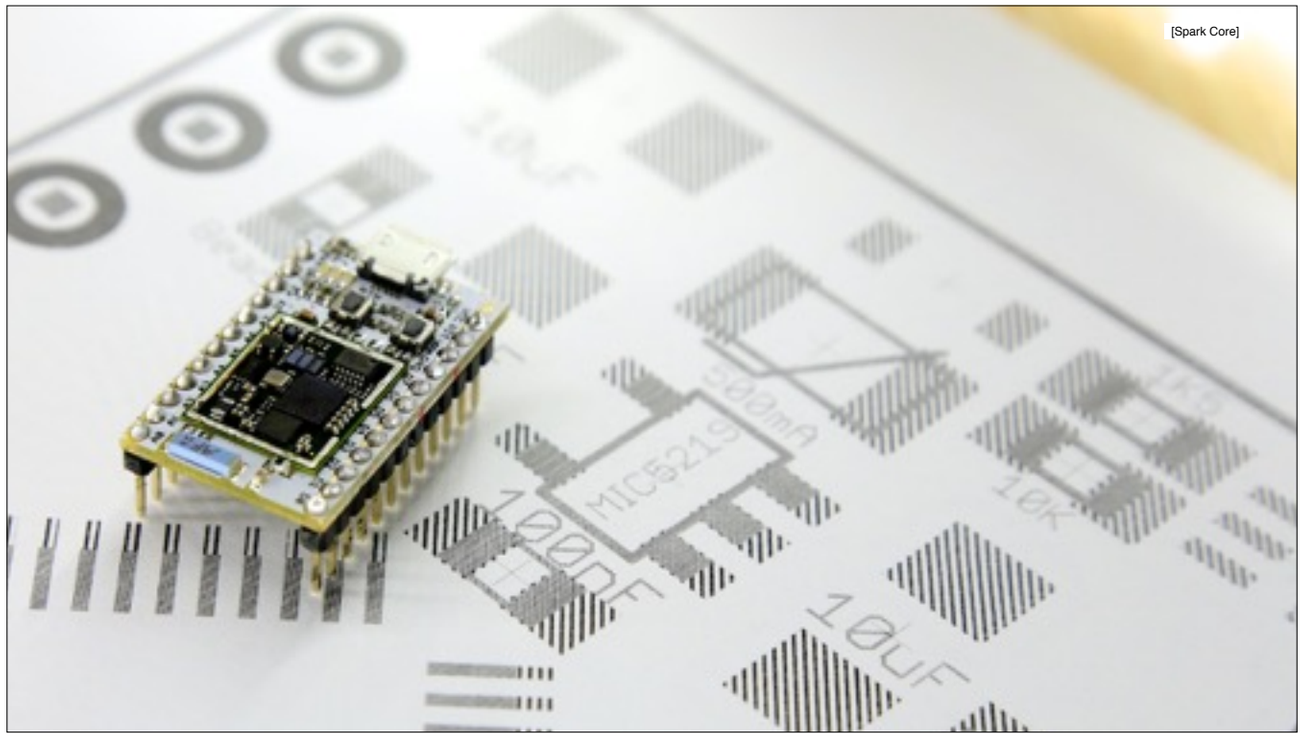
[Spark Core]

WHAT IS A SPARK CORE?

[Spark Core]



Spark OS is a complete open source operating system for cloud-connected things.



The spark core is programmed with the Arduino language and has some additional classes to connect to the cloud.

[Spark Core]

WWW.SPARK.IO

https://www.spark.io/

You can get all the information you need with code examples and other things from their website and their github account

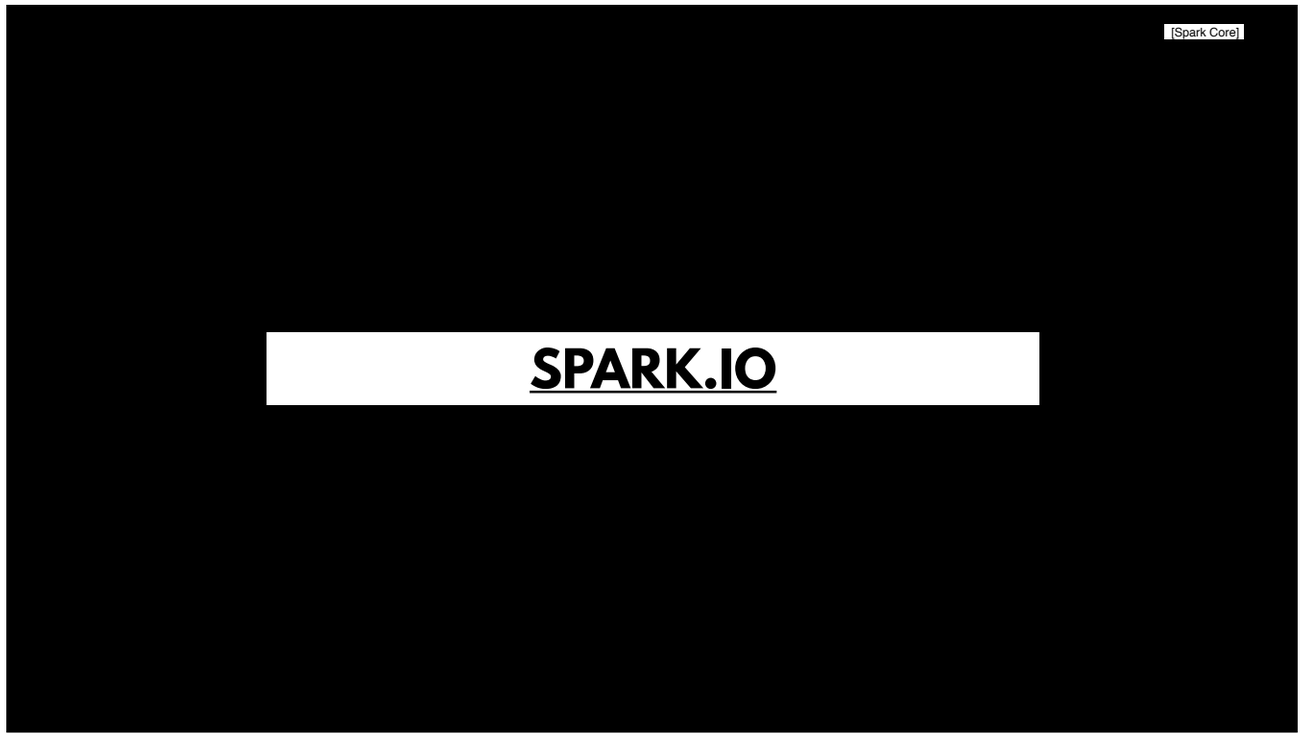
GITHUB.COM/SPARK

https://www.spark.io/ https://github.com/spark

You can get all the information you need with code examples and other things from their website and their github account

[Spark Core]

HOW TO GET IT?



Cost: 39$

If you order make a request to don’t send it with USPS. Use UPS or FedEx

[Spark Core]

DEMO

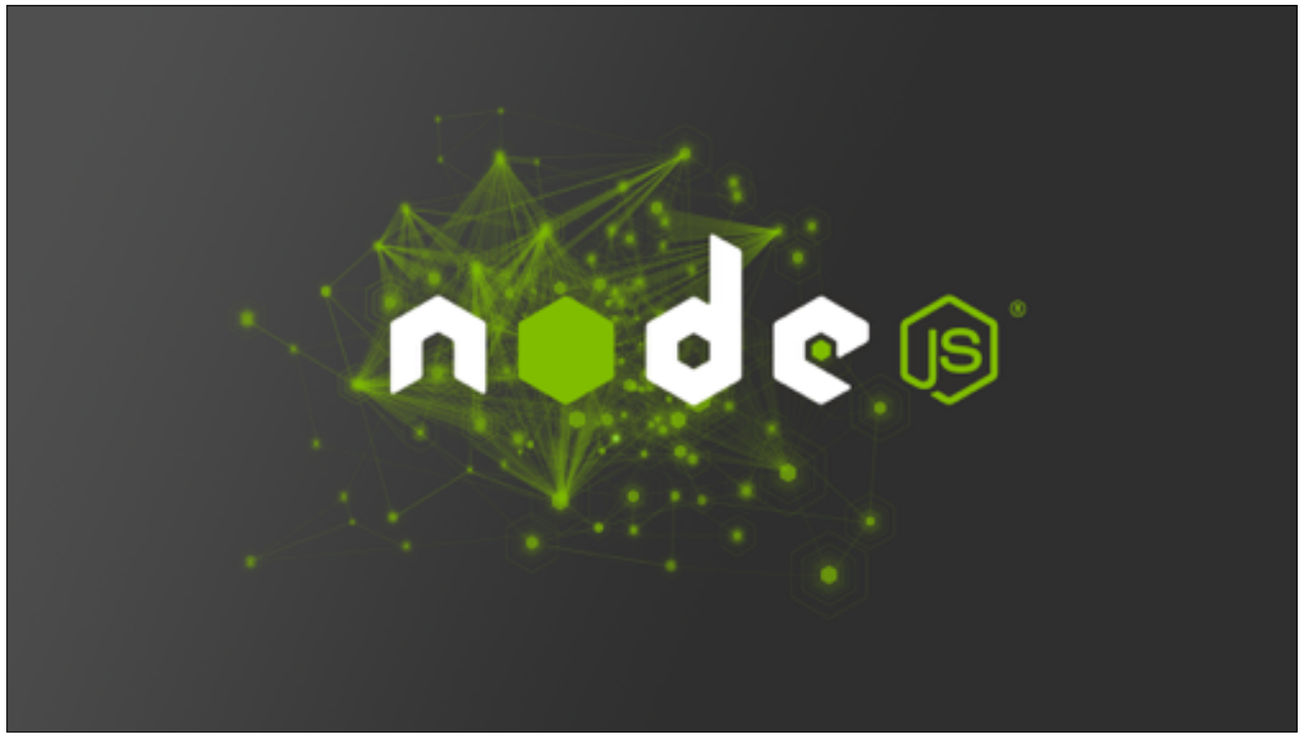
PREREQUISITES

PREREQUISITES

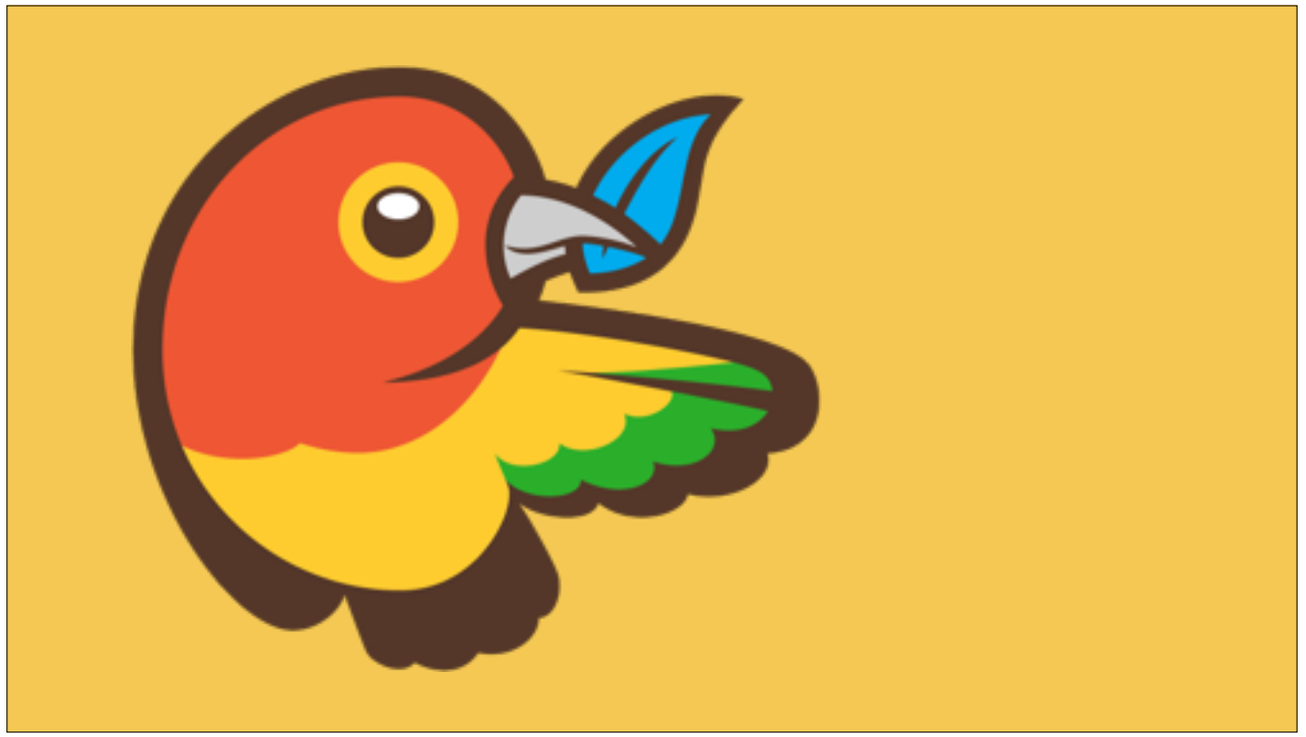
- Terminal or CMD
- Homebrew (OSX)
- (Git installed via Homebrew XCode command line tools)
- Node.js
- Bower/Grunt/http-server
- spark-cli
- spark account
- CoolTerm (OSX optional)

Package Manager for OS X

version control system Git



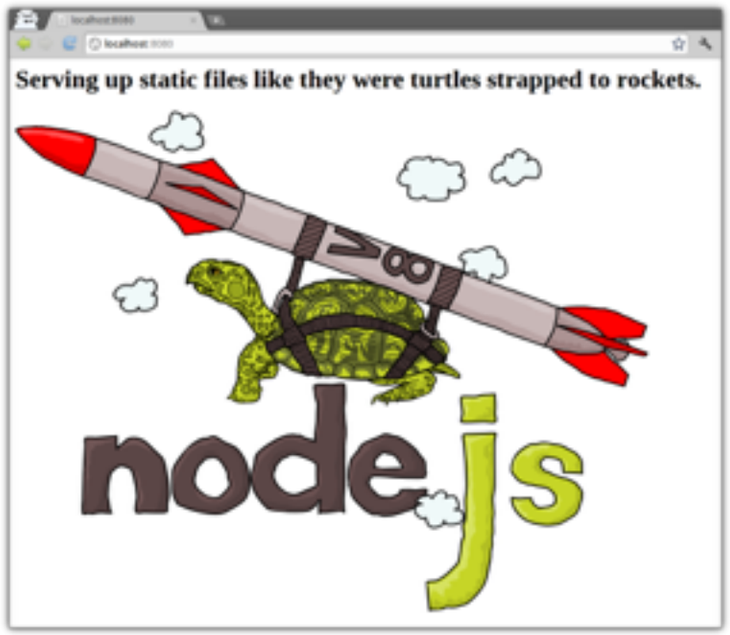
Server side JS Nodejs



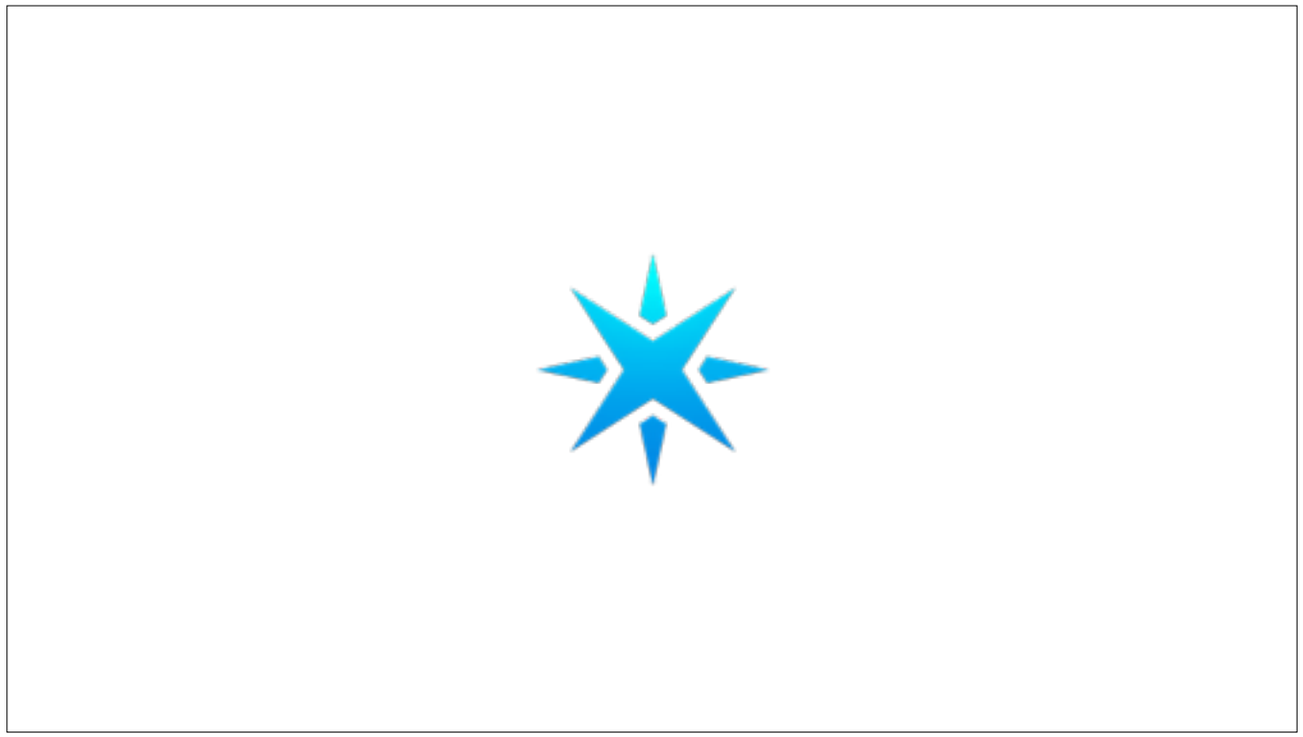
library package manager Bower.io

js task runner Grunt

[Spark Core]

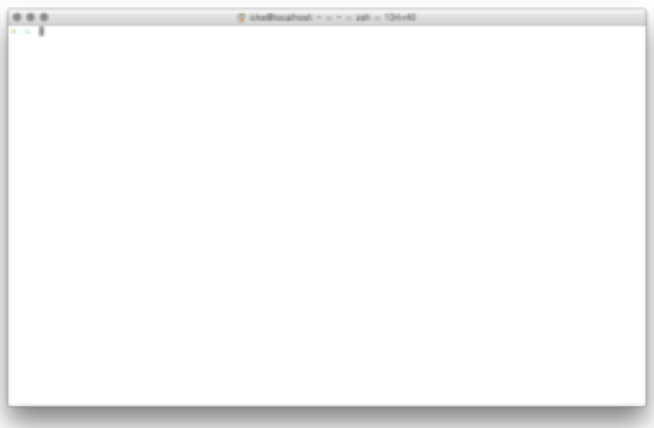


a simple server to run things like ajax locally http-server



spark CLI and an account

create an account https://www.spark.io/signup



```
cd ~/Desktop

curl -L https://raw.githubusercontent.com/FH-Potsdam/2014-2015-WiSe-spark-core-workshop/master/install.sh | sh
```

Easiest way to install. !Always read scripts like these.

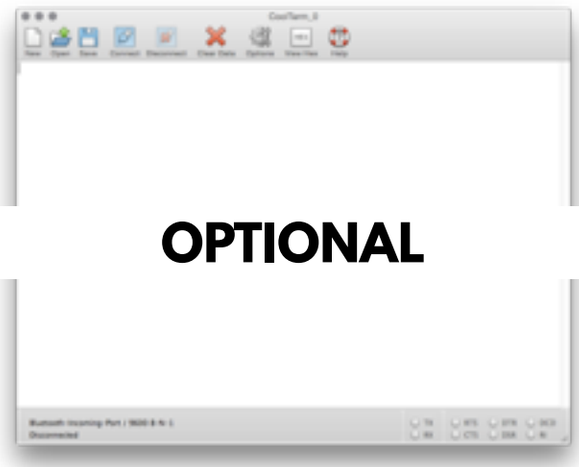
Unfortunately the windows user must install these components manually

WINDOWS

- install Node.js http://nodejs.org/
- install Git http://git-scm.com/
- run in CMD:

```
npm install -g bower grunt-cli spark-cli http-server
```

- install Github client https://windows.github.com/
- clone the repo https://github.com/FH-Potsdam/2014-2015-WiSe-spark-core-workshop



CoolTerm for serial communication / wifi setup (used cores)

Windows -> Putty

[Spark Core]

```
brew install caskroom/cask/brew-cask

brew cask install coolterm

# will be linked to ~/Applications/CoolTerm.app
```

This is the easy and nerdy way to install CoolTerm

SPARK-CLI

[command line interface]

[Spark Core]

```
➜  ~  spark

Welcome to the Spark Command line utility!
https://github.com/spark/spark-cli

Usage: spark <command_name> <arguments>
Common Commands:

    setup, list, call, get, core, identify, flash, subscribe
    compile, monitor, login, logout, help

Less Common Commands:
    cloud, config, function, keys, serial, udp, variable
    webhook

For more information Run: spark help <command_name>

➜  ~
```

[Spark Core]

SETUP, CLAIM & BLINK

[Spark Core]

SETUP & CLAIM

[Spark Core]

FRESH SPARK?

Listening mode flashing blue

[Spark Core]

```
spark login

spark setup
```

Wifi Credentials?

[Spark Core]

USED SPARK?

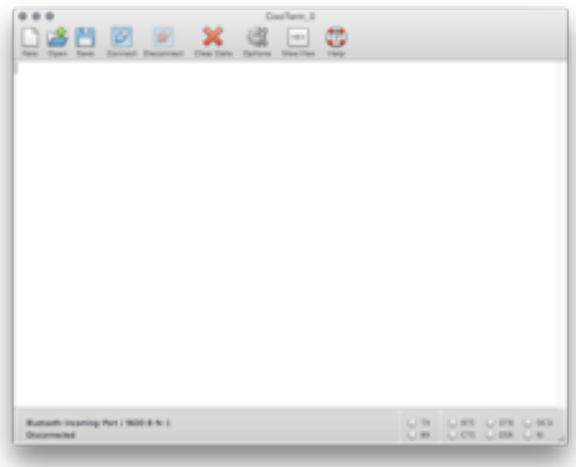
[Spark Core]

LISTENING MODE

(Hold the "Mode" button for 3 seconds, then the LED should be flashing blue. To remove all WIFI settings hold the "Mode" button 10 seconds

```
spark setup wifi
```

[Spark Core]



[Spark Core]

COOLTERM

Settings & Usage

- Baudrate: 9600
- Data Bits: 8
- Parity: none
- Stop Bits: 1
- w: Set up your Wi-Fi SSID and password
- i: ("i" as in identify) Read out the Spark Core ID

[Spark Core]

MODES

- **Blinking blue**: Listening for Wi-Fi credentials
- **Solid blue**: Getting Wi-Fi info from app
- **Blinking green**: Connecting to the Wi-Fi network
- **Blinking cyan**: Connecting to the Spark Cloud
- **Blinking magenta**: Updating to the newest firmware
- **Breathing cyan**: Connected!

See an animation

[Spark Core]

BLINK

[Out of the Box]

[Spark Core]

```
spark function call YOUR_CORES_NAME_OR_ID digitalWrite "D7/HIGH"
```

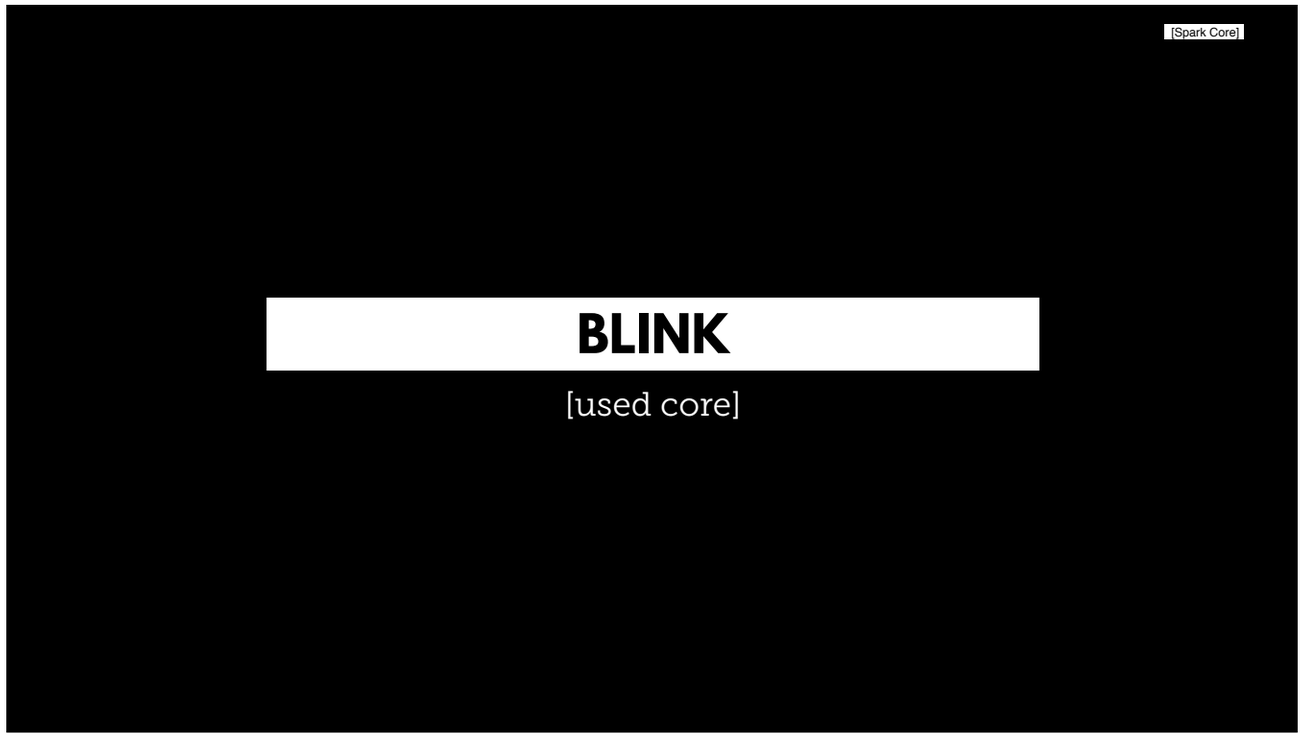
tinker

TINKER

docs.spark.io/tinker

http://docs.spark.io/tinker/

[Spark Core]



factory reset

[Spark Core]

```
spark flash [CORE ID OR NAME] tinker
```

[Spark Core]

FLASH TINKER

- Try a factory reset. Hold down both buttons, then release the RST button, while holding down the MODE button. The LED should begin flashing yellow. Continue holding down the MODE button until you see the Core change from flashing yellow to flashing white. Then release the button. The Core should begin after the factory reset is complete. link

To reflash Tinker from within the app:

- iOS Users: Tap the list button at the top left. Then tap the arrow next to your desired Core and tap the "Re-flash Tinker" button in the pop out menu.
- Android Users: With your desired Core selected, tap the options button in the upper right and tap the "Reflash Tinker" option in the drop down menu.re should begin flashing blue after the factory reset is complete.

http://docs.spark.io/connect/#appendix-factory-reset

[Spark Core]

```
spark function call YOUR_CORES_NAME_OR_ID digitalWrite "D7/HIGH"
```

tinker

[Spark Core]

CODE EXAMPLES

[firmware & web]

[Spark Core]

SRC/*.*

My examples (hook up my cores)

[Spark Core]

SPARK-HELPER/*.*

https://github.com/jflasher/spark-helper

https://github.com/jflasher/spark-helper

[Spark Core]

interface.fh-potsdam.de/spark-core/helper

get the credentials from me

SPARK-AJAX/*.*

http://fmz.pictor.uberspace.de/2014-2015-WiSe-spark-core-workshop/spark-ajax/

http://fmz.pictor.uberspace.de/2014-2015-WiSe-spark-core-workshop/spark-ajax/

[Spark Core]

WEB-INTERFACE/*.*

https://github.com/suda/spark-web-interface/

http://suda.github.io/spark-web-interface/

[Spark Core]

suda.github.io/spark-web-interface/

fh-potsdam.github.io/spark-web-interface/

http://suda.github.io/spark-web-interface/

http://fh-potsdam.github.io/spark-web-interface/

[Spark Core]

SPARK-CLIENT-SIDE/*.*

```
error!
```

[Spark Core]

[Spark Core]

TRIX & HINTS

[Spark Core]

SPLIT DEVELOPMENT

hardware prototyping (with Arduino)
cloud connection (with Spark Core)

[Spark Core]

DON’T WRITE BLOCKING CODE

[Spark Core]

```
delay(5000); // bad idea
```

[Spark Core]

```
// good idea
int update = 5000;
void setup(){}
void loop() {
    if(millis() > update){
        update+= millis();
        // do something
   }
}
```

[Spark Core]

```
spark help [COMMAND]
```

[Spark Core]

```
spark list
```

[Spark Core]

```
spark monitor [VARIABLE NAME]
```

[Spark Core]

```
spark function call [CORE NAME] [FUNCTION NAME] ["DATA"]
```

spark subscribe mine

```
spark setup wifi
```

[Spark Core]

```
spark flash [YOUR_CORES_NAME] [SOURCECODE_FOLDER]
```

via cloud

[Spark Core]

```
spark compile [SOURCECODE_FOLDER or FILE]
```

cloud compile to local file

[Spark Core]

DFU MODE

(DEVICE FIRMWARE UPGRADE)

If you are wish to program a Core with a custom firmware via USB, you'll need to use this mode. This mode triggers the on-board bootloader that accepts firmware binary files via the dfu-utility.Procedure:

- Hold down BOTH buttons
- Release only the RST button, while holding down the MODE button.
- Wait for the LED to start flashing yellow
- Release the MODE button
- The Core now is in the DFU mode.

http://docs.spark.io/cli/#installing-advanced-install

http://docs.spark.io/cli/#installing-advanced-install

[Spark Core]

```
spark flash --usb firmware_XXX.bin [CORE ID or CORE NAME]
```

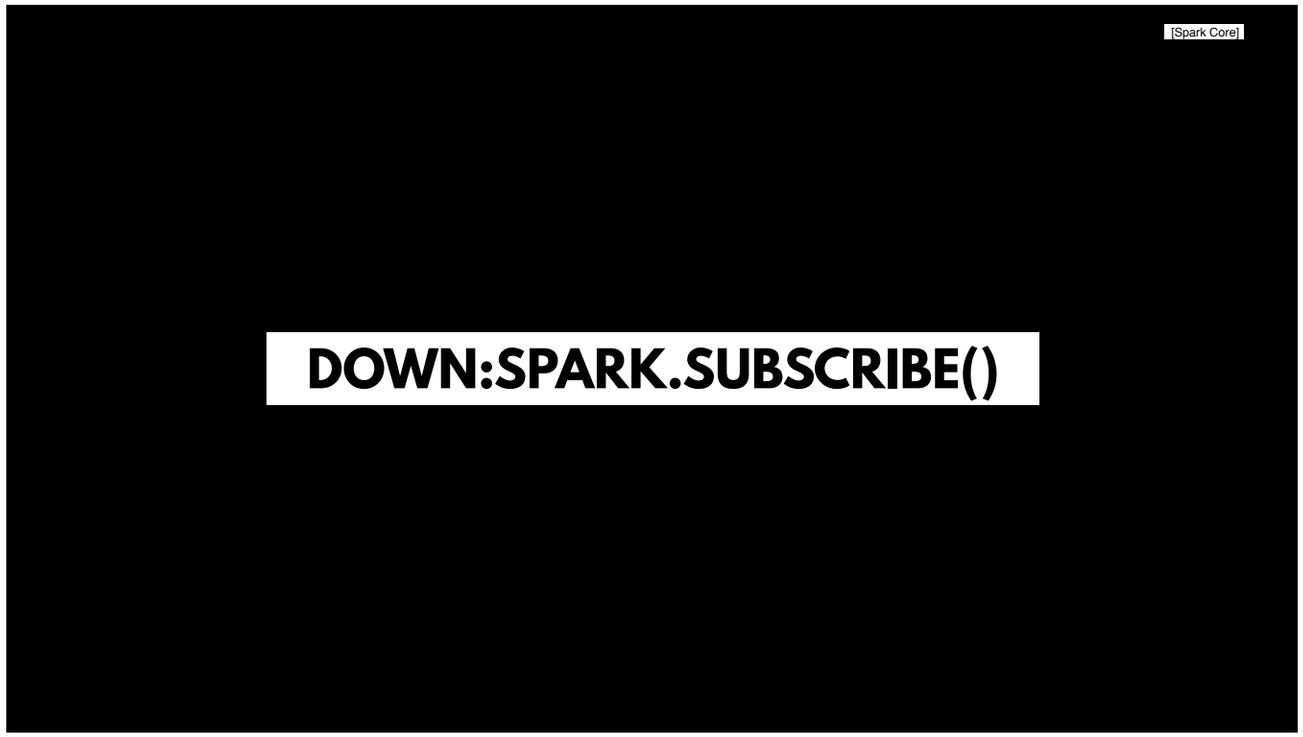
[Spark Core]

UPS & DOWNS & TODOS

[Spark Core]

UP:OUT OF THE BOX

Server, Command Line, HTML, CSS, JS



The problem was that I was publishing to many events

TODO:SPARK-SERVER

https://github.com/spark/spark-server

[Spark Core]

TODO:SPARKJS (CLIENT SIDE)

https://github.com/spark/sparkjs

[Spark Core]

TODO:SPARKJS (SERVER SIDE)

https://github.com/spark/sparkjs

[Spark Core]

Q&A

GITHUB.COM/FH-POTSDAM/2014-2015-WISE-SPARK-CORE-WORKSHOP

https://github.com/FH-Potsdam/2014-2015-WiSe-spark-core-workshop

[Spark Core]

HANDS ON