



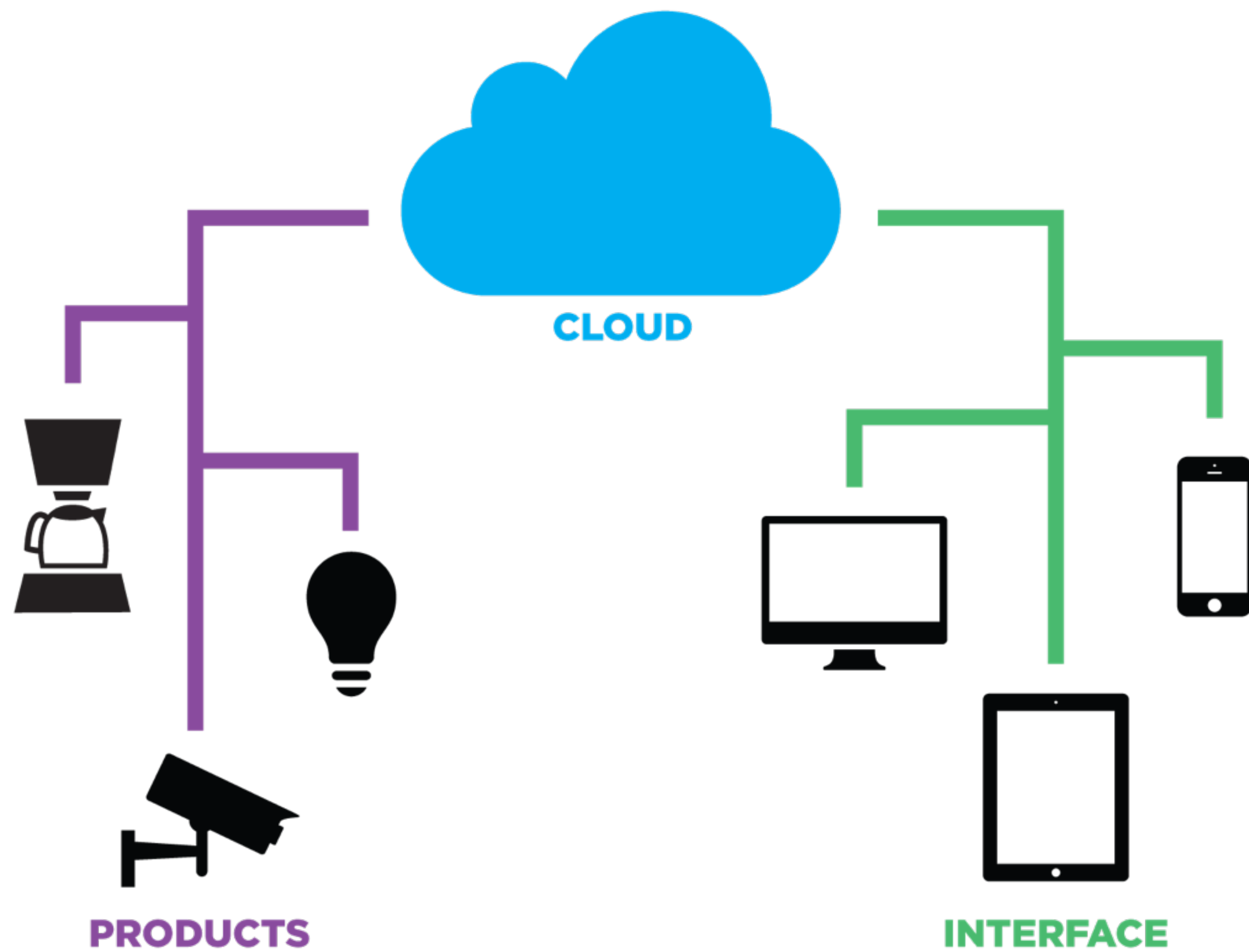
SPARKCORE WORKSHOP

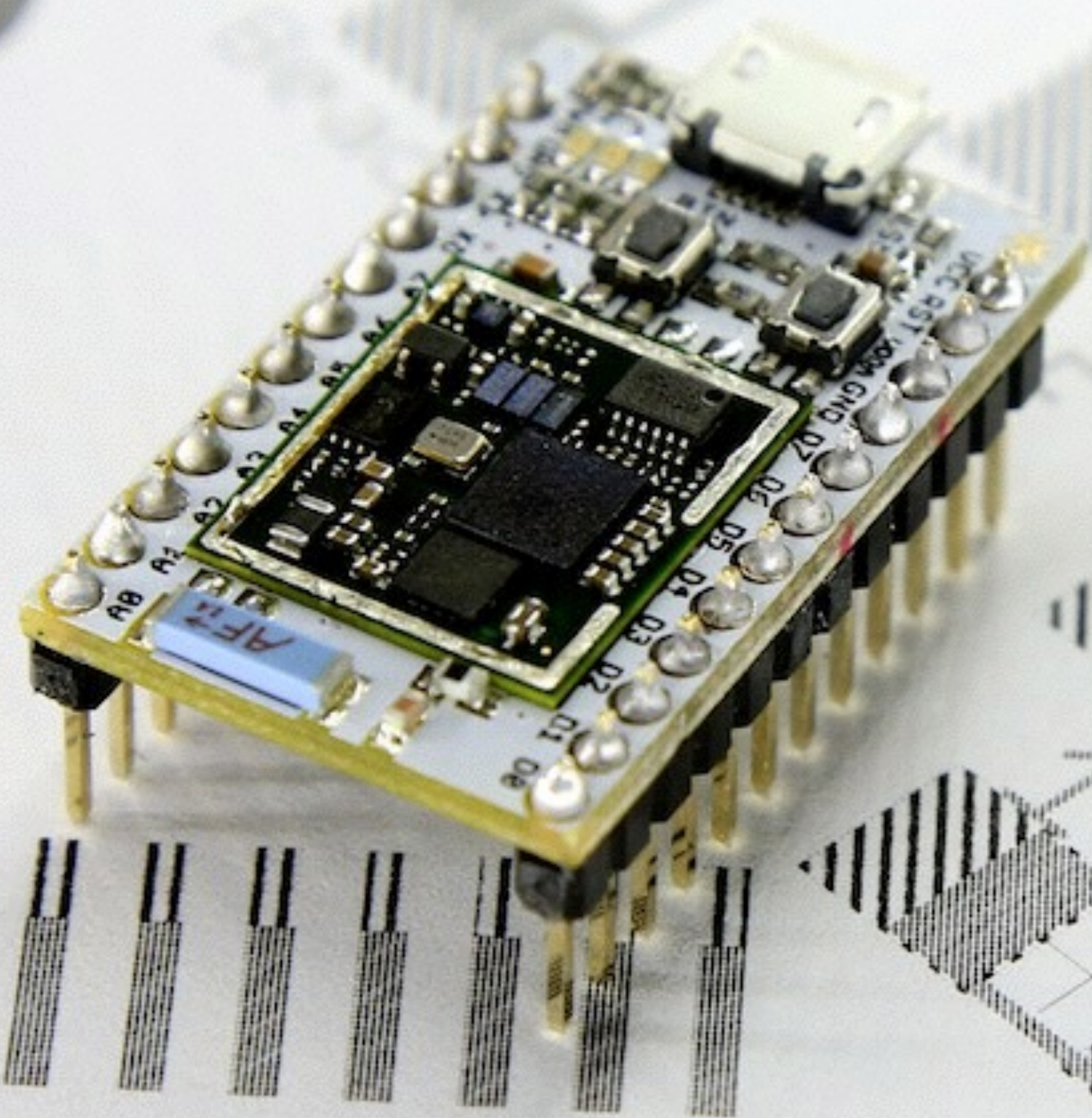
Fabian Morón Zirfas
University of Applied Sciences Potsdam (Germany)
2014

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

WHAT IS A SPARK CORE?





WWW.SPARK.IO

GITHUB.COM/SPARK

HOW TO GET IT?

SPARK.IO

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)

Homebrew

The missing package manager for OS X

English

Fork me on GitHub

Homebrew installs **the stuff you need** that Apple didn't.

```
$ brew install wget
```

Homebrew installs packages to their own directory and then symlinks their files into `/usr/local`.

```
$ cd /usr/local
$ find Cellar
Cellar/wget/1.15
Cellar/wget/1.15/bin/wget
Cellar/wget/1.15/share/man/man1/wget.1

$ ls -l bin
bin/wget -> ../Cellar/wget/1.15/bin/wget
```

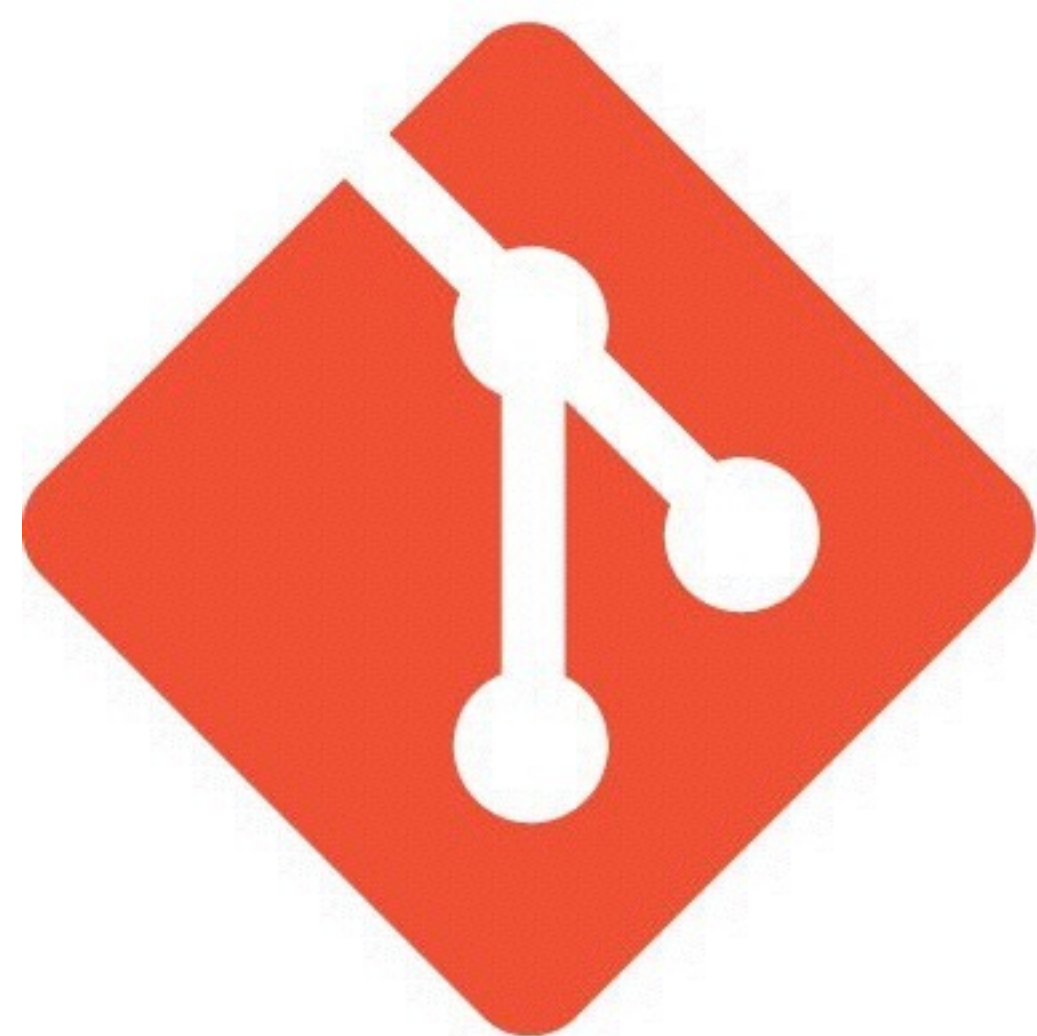
Homebrew won't install files outside its prefix, and you can place a Homebrew installation wherever you like.

Trivially create your own Homebrew packages.

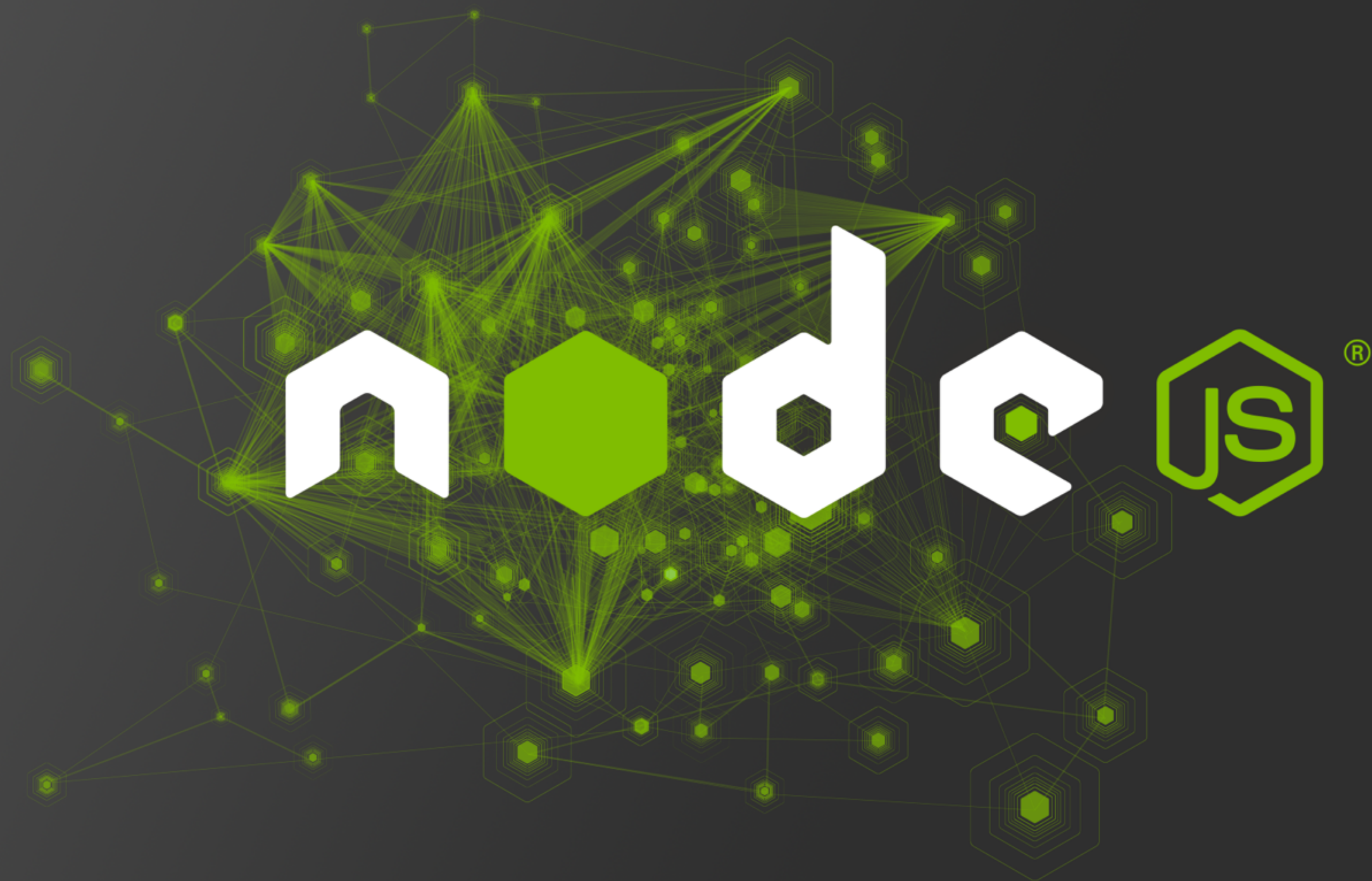
```
$ brew create http://foo.com/bar-1.0.tgz
Created /usr/local/Library/Formula/bar.rb
```

It's all git and ruby underneath, so hack away with the knowledge that you can easily revert your modifications and

```
$ brew edit wget # opens in $EDITOR!
```

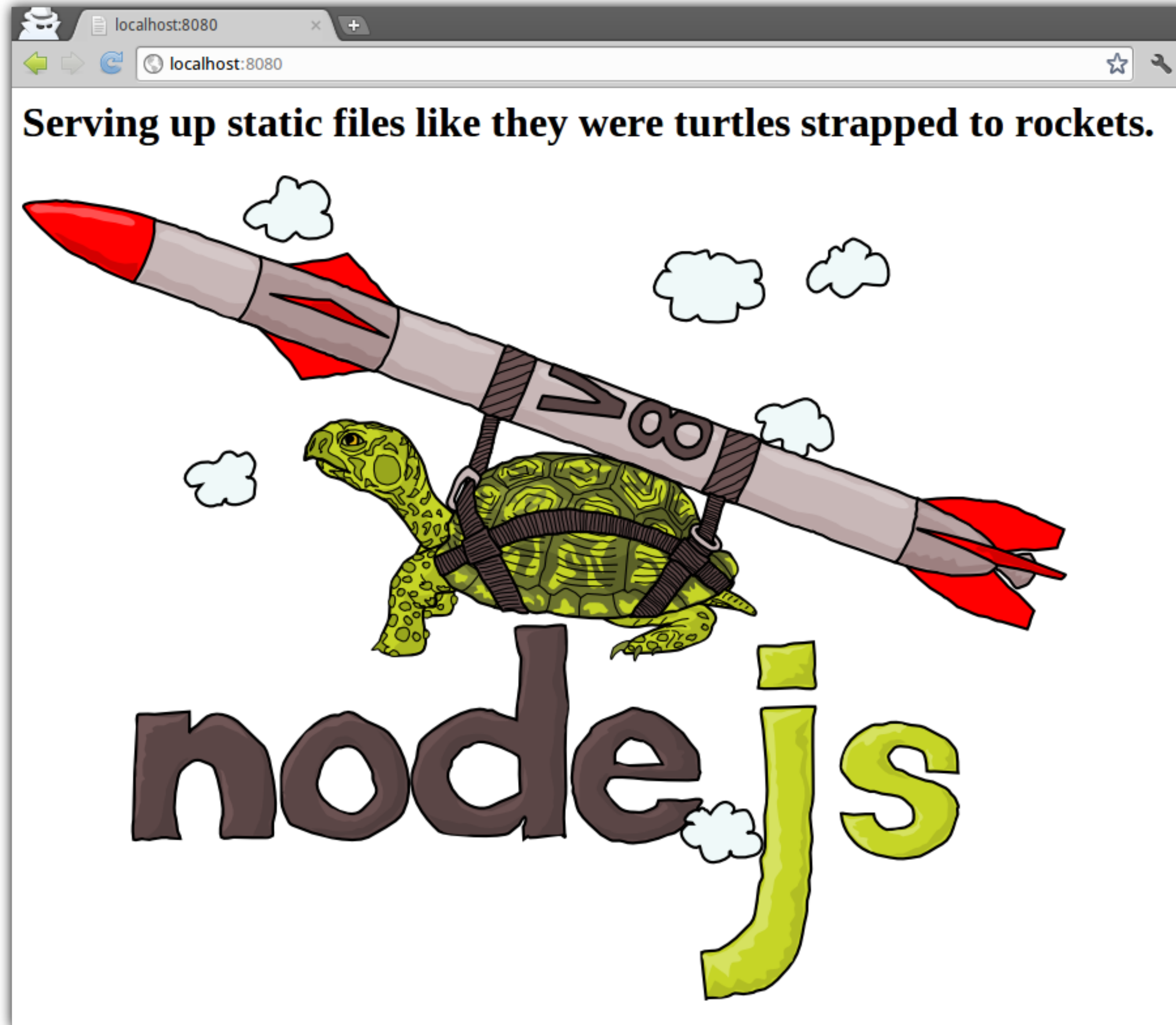



git



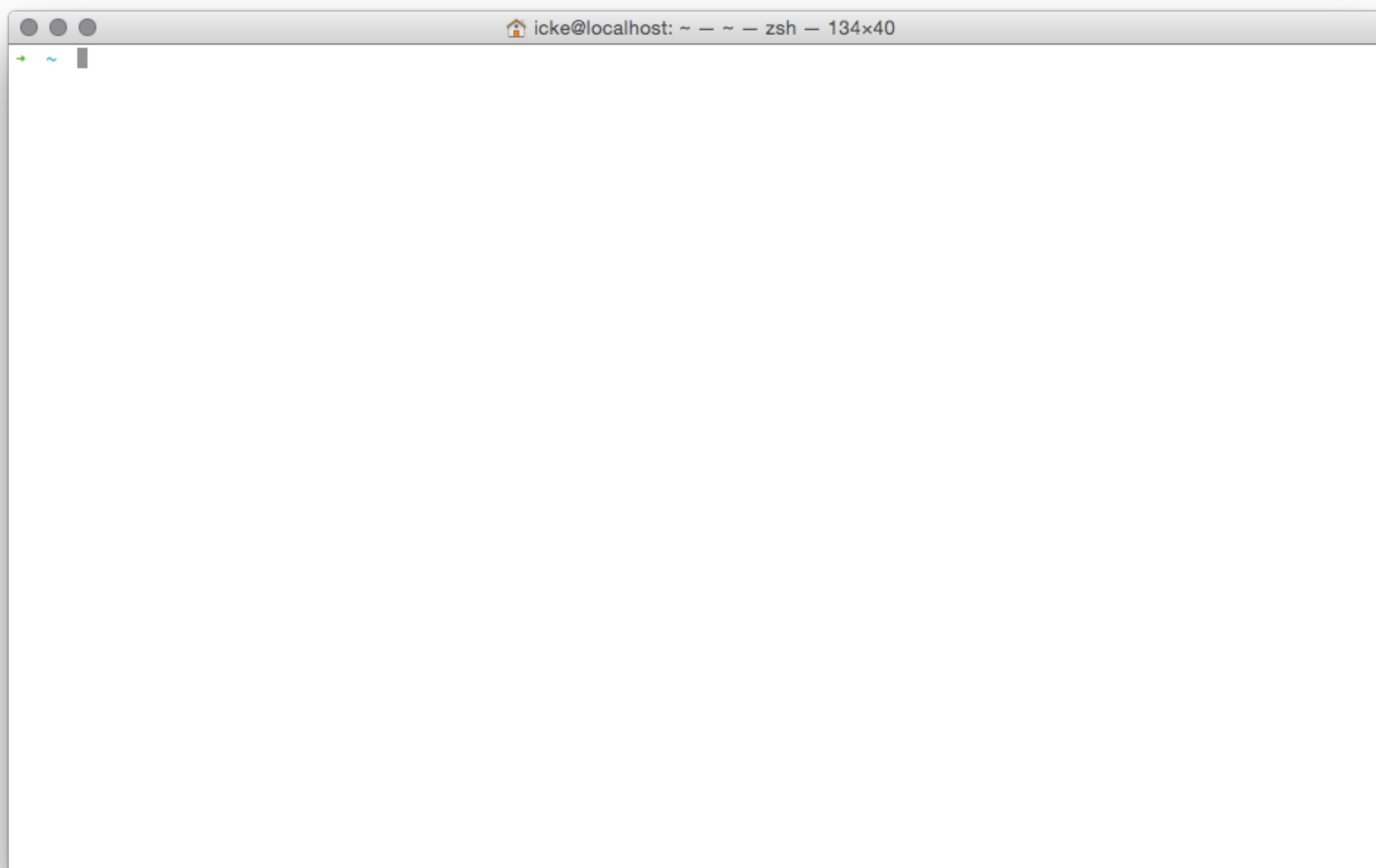








<https://www.spark.io/signup>



```
cd ~/Desktop
```

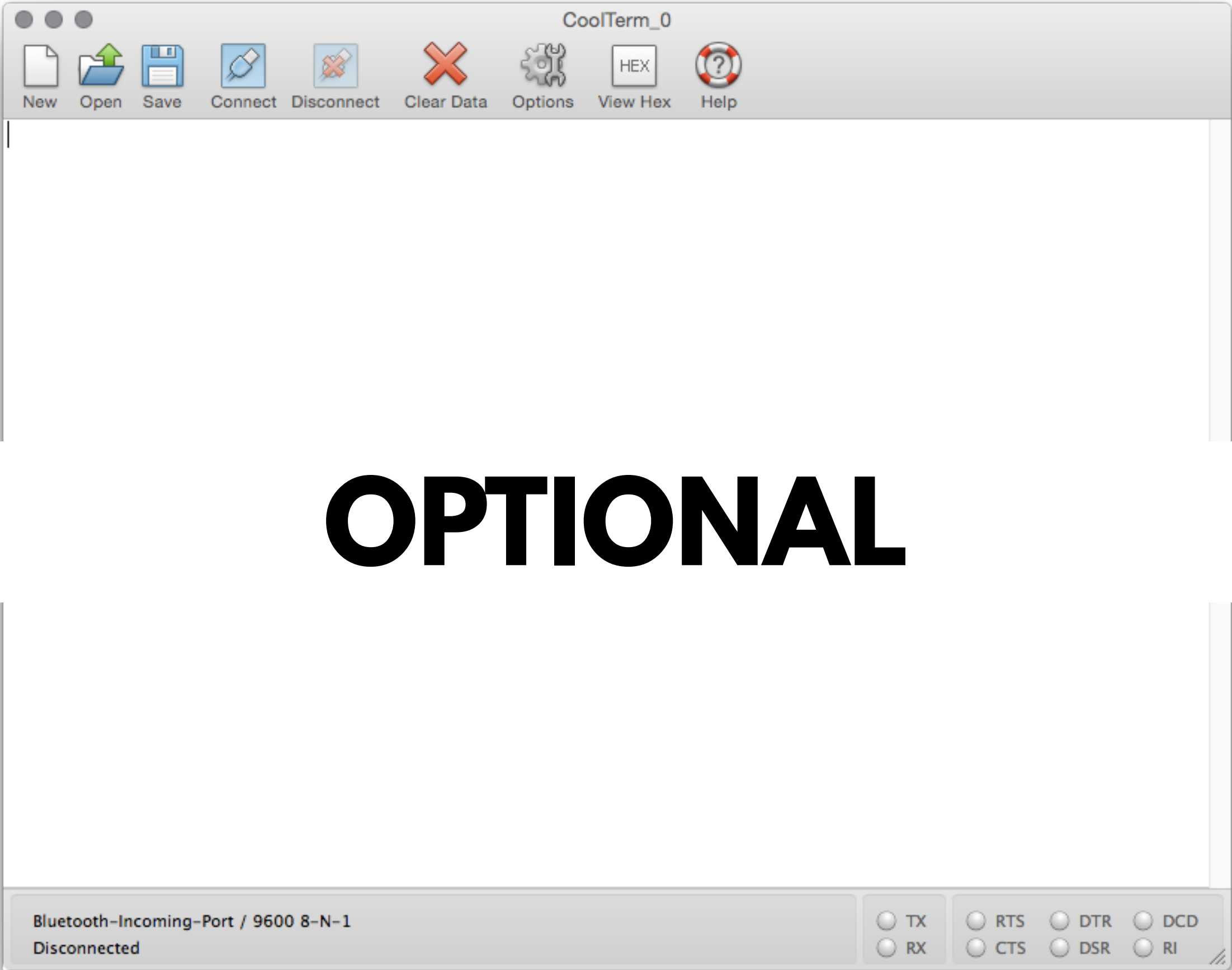
```
curl -L https://raw.githubusercontent.com/FH-Potsdam/2014-2015-WiSe-spark-core-workshop/master/install.sh | sh
```

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>



OPTIONAL

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

```
brew cask install coolterm
```

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

SPARK-CLI

[command line interface]

```
icke@localhost: ~ -- zsh -- 80x24
→ ~ 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>

→ ~ █
```


SETUP, CLAIM & BLINK

SETUP & CLAIM

FRESH SPARK?

SETUP VIA USB

LED flashing blue on connection
(listening mode)

spark login

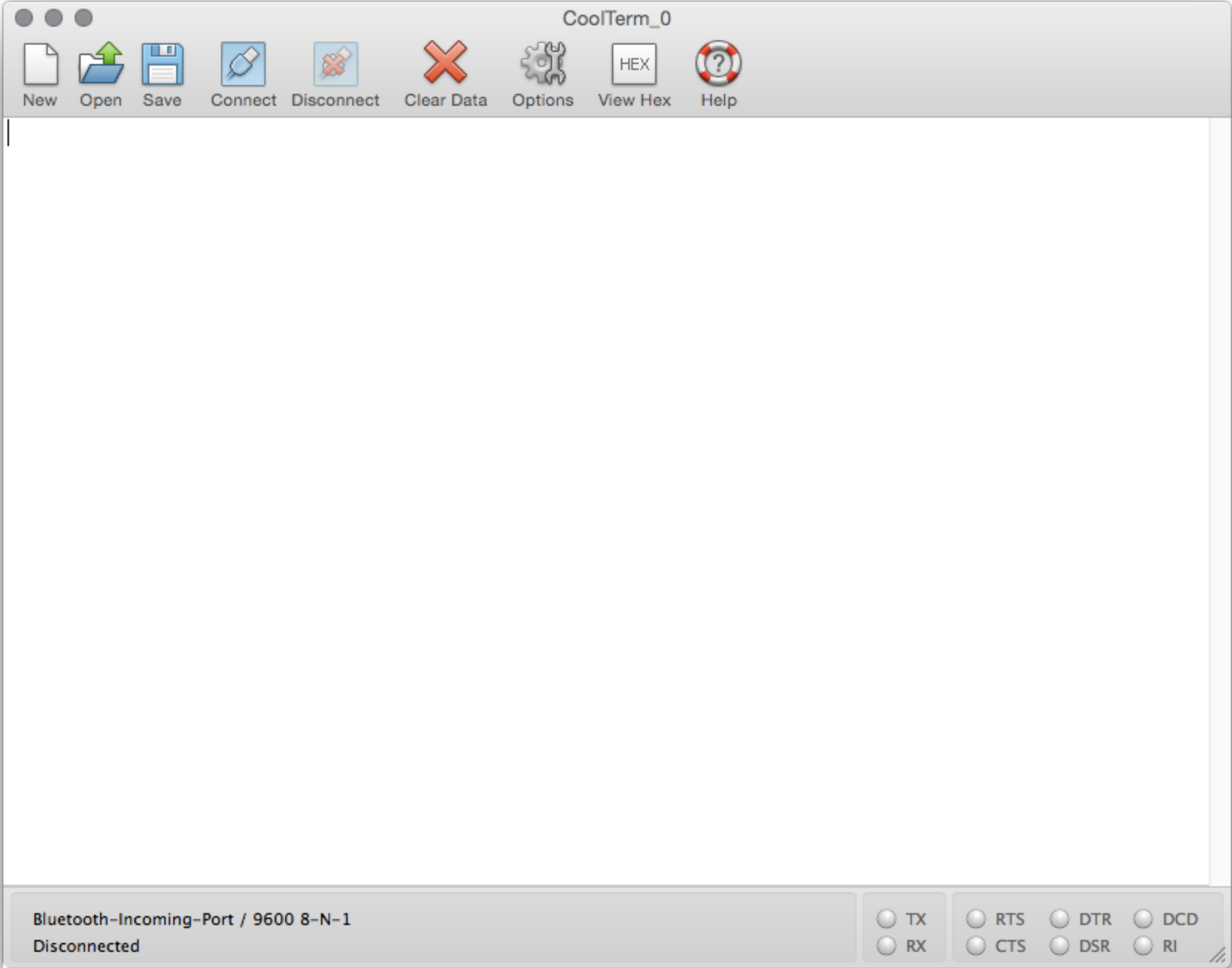
spark setup

USED SPARK?

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



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

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](#)

BLINK

[Out of the Box]

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

TINKER

docs.spark.io/tinker



BLINK

[used core]

spark flash [CORE ID OR NAME] tinker

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. It should begin flashing blue after the factory reset is complete.

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

CODE EXAMPLES

[firmware & web]

SRC/*.*

SPARK-HELPER/*.*

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

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

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/)

WEB-INTERFACE/*.*

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

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

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

SPARK-CLIENT-SIDE/*.*

er for joy!

TRIX & HINTS

SPLIT DEVELOPMENT

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

DON'T WRITE BLOCKING CODE

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

```
// good idea

int update = 5000;

void setup(){}

void loop() {

    if(millis() > update){

        update+= millis();

        // do something

    }

}
```

spark help [COMMAND]

spark list

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

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

spark subscribe mine

spark setup wifi

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

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


DFU MODE

(DEVICE FIRMWARE UPGRADE)

If you 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.

```
brew install dfu-util
```

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

UPS & DOWNS & TODOS

UP:OUT OF THE BOX

DOWN:KNOWLEDGE NEEDED

DOWN:SPARK.SUBSCRIBE()

TODO:SPARK-SERVER

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

TODO:SPARKJS (CLIENT SIDE)

<https://github.com/spark/sparkjs>

TODO:SPARKJS (SERVER SIDE)

<https://github.com/spark/sparkjs>

Q&A

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

HANDS ON