



OPEN CONNECTIVITY
FOUNDATION®

跨平台的物联网网络的自动化开发过程

在15分钟内开发一个安全、可被认证的硬件原型

Clarke Stevens
Shaw Communications
clarke.stevens@sjrb.ca

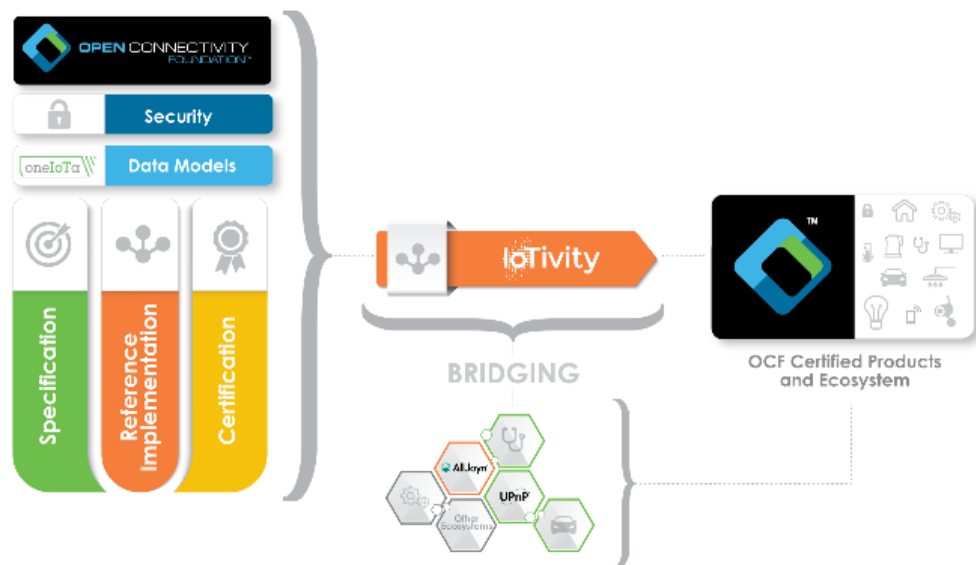
Shaw)



Open Connectivity Foundation 是什么？

- 制定互操作物联网标准的标准组织
- 从头构建的强大安全属性
- 可与现有IoT生态系统互操作
- 三大支柱
 - 国际标准协议
 - 开源代码实现
 - 自动化认证工具和国际授权实验室
- 灵活的、RESTful风格的、基于数据模型的架构

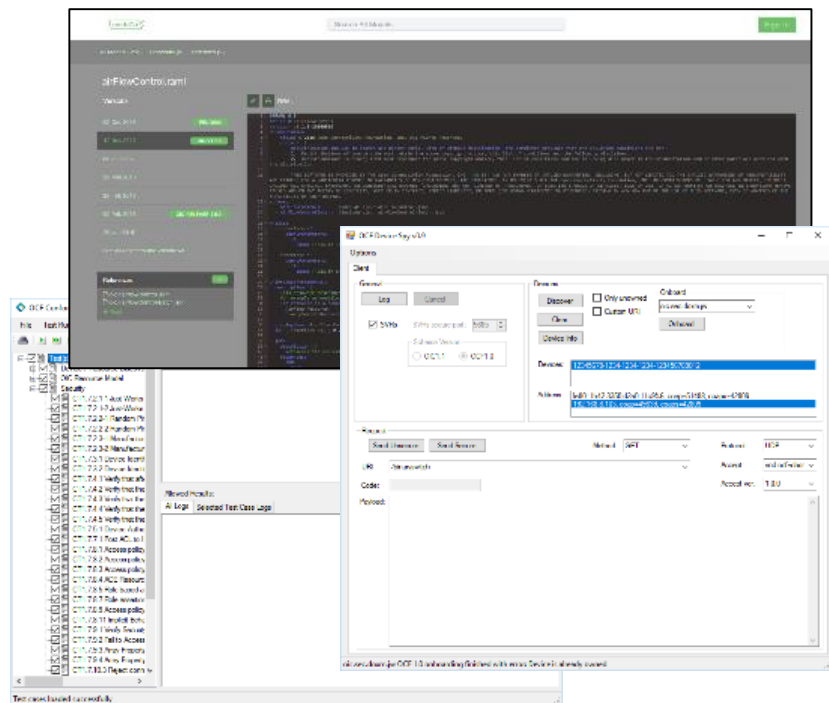
Building Blocks for Success





全套的开发工具

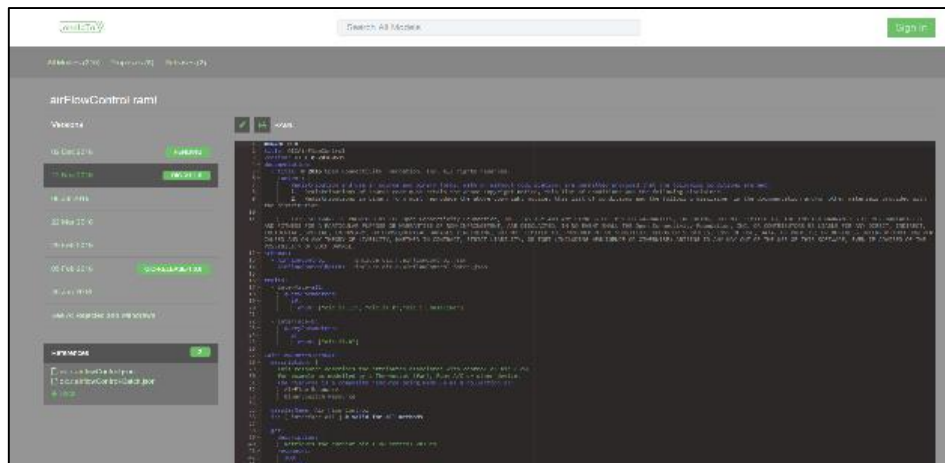
- 任何人都可使用或参与建设的在线数据模型工具（需遵循标准格式），也是建立生态系统互操作的基础。
- 数据模型驱动一切。
- 自动化工具链可创建：
 - 服务器端源代码
 - Introspection文件
 - 自动化的测试用例文件
 - 安全的设备onboarding联网文件
 - 客户端工具的用户界面
- 多种测试和开发工具





oneloTa (oneiota.org)

- oneloT是一个资源数据模型的集成开发环境和过程管理工具，是OCF的核心。
 - 在oneloTa的编辑窗口中输入OCF原子资源模型（译者注：原子资源指最基本的简单资源）
 - 较复杂的资源可由原子资源组成。
 - 提交完成的资源模型，OCF（或其他合作组织）按照审核流程进行审核。
 - 在oneloTa中，其他组织也使用和OCF同样的审核流程。
 - OCF和其他生态系统的数据模型之间的映射，同样需在oneloTa提交。
 - 资源模型用于搭建标准协议、设备、源代码、图形界面、PICS文件和桥接功能等。





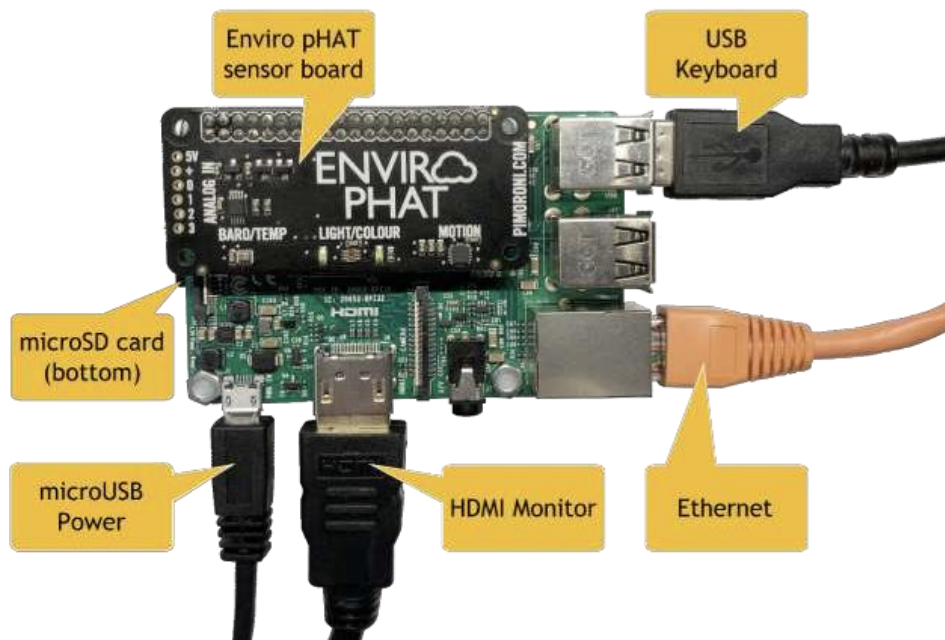
oneIoTa 是你需要的所有资源的来源

- 请至 <http://www.oneiota.org>
- 搜索任何你需要的资源
- 参考和使用这些描述文件，定义你自己的设备描述文件
- 每种设备都包含一系列的资源
- 前述所有工具都把oneIoTa作为资源的来源



准备硬件

- 在树莓派的GPIO header上安装AutomationPhat
- 插入microSD卡
- 连接网络
- 连接键盘和显示器（或SSH）
- 接上电源





准备树莓派的SD卡

- Go through the NOOBS process to install the headless Raspian Stretch Lite located here: <https://www.raspberrypi.org/downloads/raspbian/>
- Enable ssh: copy an empty file called "ssh" (no extension) to the SD card root directory (<https://github.com/openconnectivity/Sample-Raspberry-Pi-Code/blob/master/ssh>)
- Optionally set up a Wi-Fi connection by copying wpa_supplicant.conf to the SD card root directory and edit for your Wi-Fi (https://github.com/openconnectivity/Sample-Raspberry-Pi-Code/blob/master/wpa_supplicant.conf)
- 使用电脑控制树莓派很简单，知道树莓派的IP地址即可。通过HDMI连接树莓派和显示器，IP地址会显示在启动过程的接近结尾处。
- 通过插入SD卡并接上电源来启动树莓派，你就可以记录IP地址了。
- 使用显示器和键盘，或ssh (Putty on Windows, or the Go menu on Mac will give you an SSH connection) 进入树莓派控制界面。
 - 默认登录信息 : user: pi, password: raspberry



搭建开发环境

- 安装开发环境
 - 切换到home目录下: `cd ~`
 - 下载并配置IoTivity-Lite: `curl https://openconnectivity.github.io/IOTivity-Lite-setup/install.sh | bash`
 - 下载并配置IoTivity(可选): `curl https://openconnectivity.github.io/IOTivity-setup/install.sh | bash`
 - 下载并安装工程配置脚本: `curl https://openconnectivity.github.io/Project-Scripts/install.sh | bash`
 - 下载树莓派例子 (在需要选择输入y/n的时候请输入y): `curl https://openconnectivity.github.io/Sample-Raspberry-Pi-Code/pi-boards/install.sh | bash`
 - 设置环境变量: `reboot` 或者 `source ~/.bashrc`



创建一个设备 (page 1)

- 创建开发目录:
 - `cd ~`
 - `mkdir workspace`
 - `cd workspace`
- 创建一个OCF工程:
 - `create_project.sh myexample`
 - `cd myexample`
- 我们将从预建的例子开始创建设备:
 - `cp ~/Sample-Raspberry-Pi-Code/IoTivity-lite/example/setup.sh ./`
 - `./setup.sh`



创建一个设备 (page 2)

- 自动生成代码文件，内省文件和安全文件：
 - `gen.sh`
- 构建可执行的工程：
 - `build.sh`
- 设置安全模式为 “ready for owner transfer method” (RFOTM):
 - `reset.sh`
- 运行程序：
 - `run.sh`
- You have successfully build an OCF device and it is ready to onboard!



使用OTGC让设备联网并控制设备

- 在安卓设备上安装OTGC (Onboarding Tool Generic Client):
 - 下载链接：
<https://github.com/openconnectivityfoundation/development-support/tree/master/otgc>
 - 安装成功后打开app
- 发现上线设备：
 - 点击环形箭头按钮
- 关联设备
 - 点击“+”按钮将设备进行关联
- 获取控制界面
 - 点击齿轮按钮获取控制界面
- You have successfully onboarded and controlled your OCF Device!



我们刚才做的是

- 从编写OCF设备配置文件开始:
 - `edit_config.sh`
- 编写OCF 输入文件:
 - `edit_input.sh`
- 编写OCF设备代码， 内省文件和安全文件：
 - `edit_code.sh`
 - `ls`
- 使用OTGC发现设备， 设备上线并进行控制：
 - `OTGC`



创建一个更复杂的设备 (page 1)

- 创建一个OCF工程:
 - `cd ~/workspace`
 - `create_project.sh automationphatlite`
 - `cd automationphatlite`
- 我们使用另一个预先创建的例子:
 - `cp ~/Sample-Raspberry-Pi-Code/IoTivity-lite/automation-phat/setup.sh ./`
 - `./setup.sh`
- 运行`setup.sh`的时候会将拷贝一些硬件接口文件
 - 查看bin文件夹下现有文件: `ls bin`



创建一个更复杂的设备 (page 2)

- 和上一个例子一样
 - 自动生成代码，内省文件，安全文件：
–gen.sh
 - 构建可执行的工程：
–build.sh
 - 设置安全模式为 “ready for owner transfer method” (RFOTM):
–reset.sh
 - 运行程序：
–run.sh
- 通过OTGC控制树莓派上的灯



现在你可以自己做了 (page 1)

- 创建一个OCF工程（名字可以任取，下面以mytestproject为例）：
 - `cd ~/workspace`
 - `create_project.sh mytestproject`
 - `cd mytestproject`
- 为你创建的设备编写OCF配置文件：
 - `edit_config.sh`
- 自动生成代码文件，内省文件和安全文件(只需运行一次除非配置文件被修改):
 - `gen.sh`



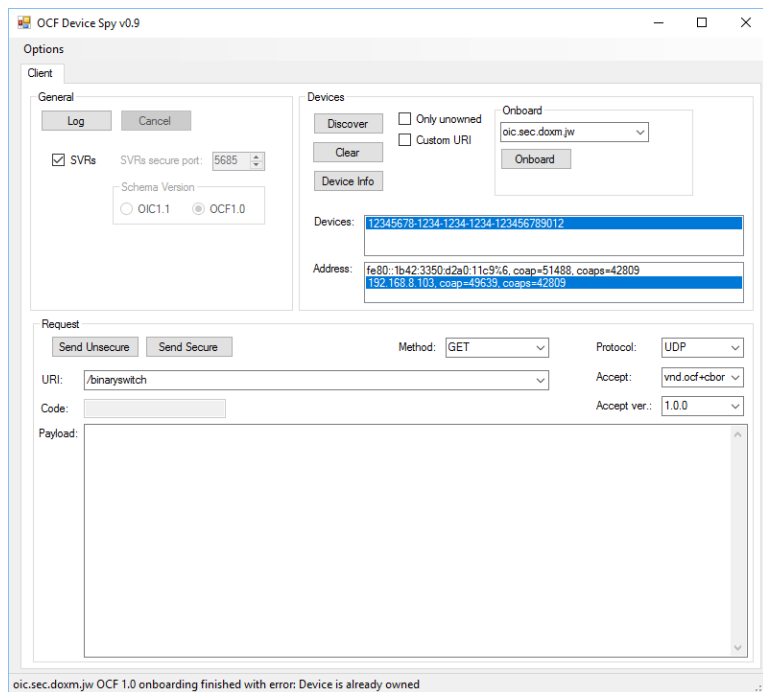
现在你可以自己做了 (page 2)

- 使用同样的开发流程，**根据需要编辑代码**：
 - Edit the code:
–[edit_code.sh](#)
 - Build the project executable:
–[build.sh](#)
 - Set the security to “ready for owner transfer method” (RFOTM):
–[reset.sh](#)
 - Run the server code on the Raspberry Pi:
–[run.sh](#)
 - Run OTGC and control the server on the Raspberry Pi



用Device Spy进行测试（OTGC之外的另一种工具）

- Device Spy是一个较基础的客户端，可以创建发送数据包的内容，Device Spy只有Windows版。
 - 点击[Discover]，发现设备。
 - 点击SVRs、选择Address、再点击[Onboard]按钮，使设备联网。
 - 设置resource URI、选择GET方法、并点击[Send Secure]按钮，检查resource的内容。
 - 选择POST消息并点击[Send Secure]按钮，改变resource的值（如，false或true）。
 - 现在灯的状态应该做出相应的改变。





Other useful information



Simple device description input file

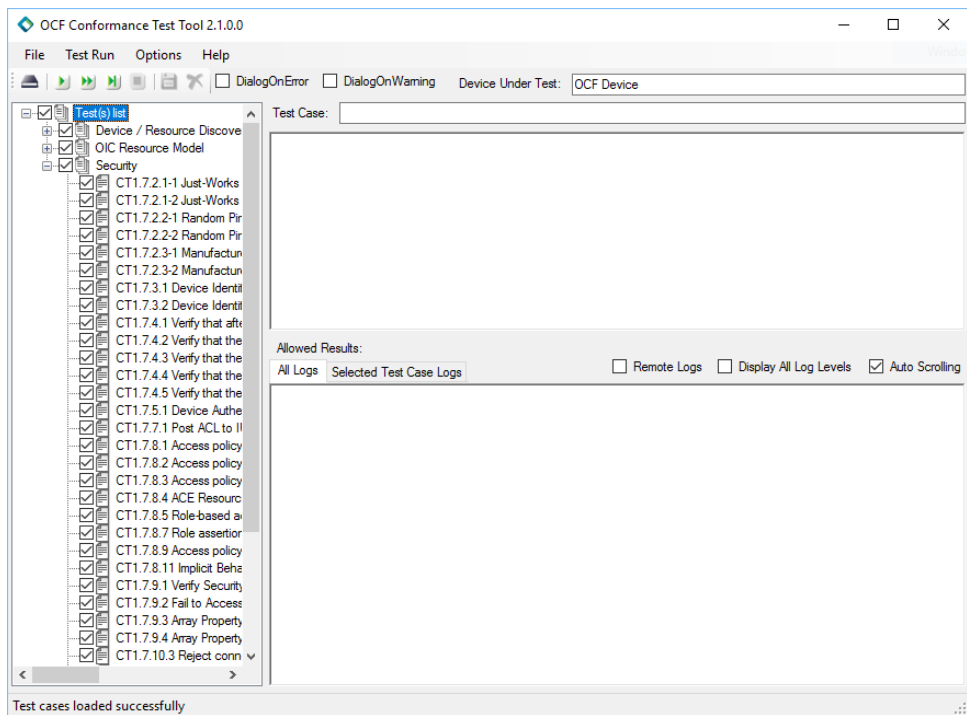
```
[
  {
    "path" : "/binaryswitch",
    "rt"   : [ "oic.r.switch.binary" ],
    "if"   : [ "oic.if.a", "oic.if.baseline" ],
    "remove_properties" : [ "range", "step", "id", "precision" ]
  },
  {
    "path" : "/oic/p",
    "rt"   : [ "oic.wk.p" ],
    "if"   : [ "oic.if.baseline", "oic.if.r" ],
    "remove_properties" : [ "n", "range", "value", "step", "precision", "vid" ]
  }
]
```



运行认证测试工具CTT (Certification Test Tool)

- CTT是一种自动化测试工具，设备必须通过CTT才能获得OCF认证。
 - 运行CTT。
 - 选择由gen.sh生成的PICS文件。
 - 选择已发现的设备及所需接口。
 - 点击play按钮开始测试。
 - 现在去喝杯咖啡。
 - 可是别去太久，因为过程中你需要把onboarding状态改为RFOTM几次。
 - 通过CTT测试后，OCF认证实验室会协助你获得官方认证。

只对OCF成员公司开放！





What to do next

- Define your own input file
- Run the tooling!
- Build it... and see if it works
- Change the code, to what you want it to do



还有

- 可使用模版修改DeviceBuilder以适配任何编程语言。
- OTGC很快就会发布Android、iOS、Linux和Windows版。
- OTGC是开源的，因此你可用多个操作系统下的客户端控制OCF设备。
- 有node.js版的IoTivity可用。
- 还有如下公司的开发板支持上述工具：
 - Samsung EagleEye
 - Linux emulator using Glade for UI
 - **更多开发板即将到来**



致谢及重要信息

- 本文所述的所有工具和网站，都是由致力于OCF发展的贡献者和合作商所开发。
(<http://openconnectivity.org>)
- 来自思科的Wouter van der Beek先生是OCF技术委员会的主席，该技术委员会负责关键架构设计。Wouter先生本人利用业余时间开发了DeviceBuilder以及文中所述的重要脚本。
- 德凯实验室作为OCF的合作商，开发了OTGC。这个工具的Android, iOS, Linux和Windows版本都将很快对公众开放下载。
- Comarch作为OCF的合作商，开发了CTT和DeviceSpy。这两个工具只对OCF成员开放。
- 更多信息请参考：
<https://github.com/openconnectivity/Project-Scripts/blob/master/One%20Page%20Complete%20Setup%20Instructions.md>



OPEN CONNECTIVITY
FOUNDATION®