

我写本文的原因是我目前工作的小组要转出北京，我要重新在北京找个工作。我的工作经历都是在做无线通信，我的简历里面侧重也都是这个方面的内容。但实际上我的软件开发的能力我个人觉得也不错，所以这篇自述主要是讲一下我个人软件开发方面做过的项目、体会，来展示一下我软件开发的素养。我的软件技术栈主要是 Python 和 Web 相关，新工作我觉得也可以拓展到这个软件开发领域。

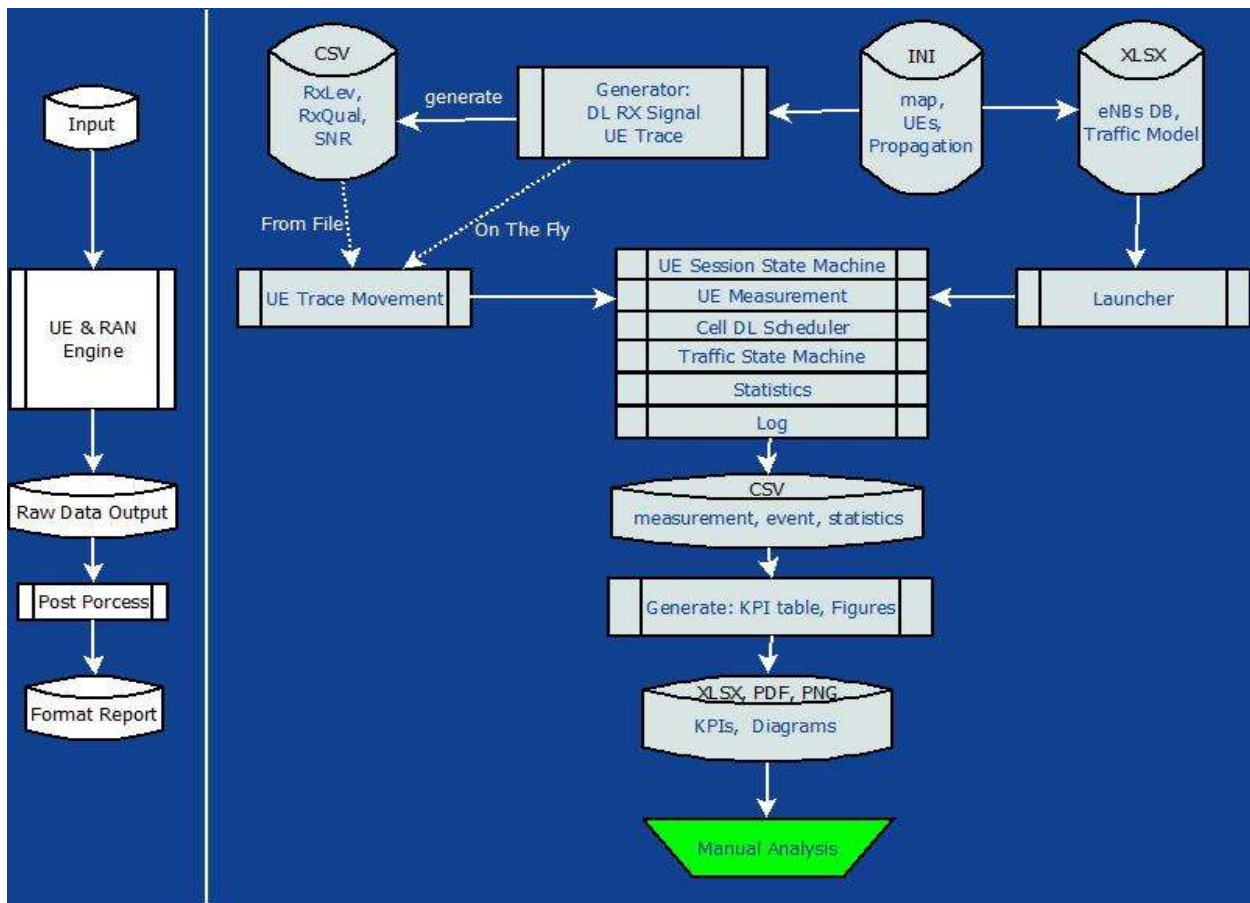
现在看来也许是幸运的，我过往的软件开发经历主要是一些个人项目。一个软件从头到尾，原因、目标、需求，设计、开发和测试都是我一个人做。这种经历锻炼我向一个全栈工程师发展。

## 项目一：一个 LTE 的仿真器，目的是评测基站的 load balance 算法。

背景：我所在的组有一部分工作是利用另外一个组开发的一个仿真器软件执行仿真以及评估结果。后来那个组停掉了开发仿真器的任务，导致我们的工作要调整去做产品需求规范的设计。在这个过程中我其实还是比较想延续仿真工作。不过当时对仿真软件怎么做一点概念也没有。所幸之前那个仿真器的源代码我可以看到，当时看了 main 函数之后，醍醐灌顶一下就明白了用软件做仿真的方法，就是下面示例的。那个仿真器是 c++ 开发的已经有一定规模，在非常有限的文档和没有传帮带的情况下，基于它继续工作还是有难度的。特别是仿真是服务于产品算法设计的，有时限要求，整个组里也只有我愿意推动这个工作，所以最后就我的一个人去继续这个任务（非常感谢我当时领导的支持）。我最终选 python，开启了我的开发之路。

```
# 仿真的原理,
for time in range(total_time):
    for each ue doAction() # 手机
    for each bs doAction() # 基站
```

设计：这里面包含两种复杂。1. 基站和手机的会话状态跃迁非常复杂，基站调度器算法复杂。输入来源于产品算法和标准规范。使用 plantuml 画了大量的 MSC 图和 state 图。2. 环境建模涉及的知识点多：地图布局，手机随机移动的设计，无线信号的强度和信噪比计算，所传数据的概率模型（这里也设计状态跃迁），需要统计的结果。由于是第一次实现仿真器，这些知识点对我来说都是新东西。下图是全局汇总。



**项目二：一个 LTE 的仿真器，目的是评测基站的切换参数自动算法。**

这个项目和上一个完全不同，它不关注数据吞吐量。它的模型需要手机和建站数量很多，仿真时间要足够长，需要出发足够多的切换来检验算法。所以主循环我改为基于事件驱动。

下面的项目 3，4，5 都是我在系统工程师角色，利用额外时间完成的，目的是提高我的日常工作效率。

**项目三：这就是我简历提到的亮点 1。4G/5G 信令消息。**

做这个项目时，我的本职工作是做系统设计。通信规范是最重要的参考，但不友好。于是在工作之余开发了这个工具。

比如手机和 LTE 基站层三信令规范：

[https://www.etsi.org/deliver/etsi\\_ts/136300\\_136399/136331/16.03.00\\_60/ts\\_136331v160300p.pdf](https://www.etsi.org/deliver/etsi_ts/136300_136399/136331/16.03.00_60/ts_136331v160300p.pdf)

(432 页到 437 页是 ASN1 的语法定义，437 到 442 页是描述信息)

用这个 tool 看到的效果参见：

<https://liuyonggang1.github.io/3GPP/asn1/36331.html#SystemInformationBlockType1>

其中 ASN1 的词法分析的灵感来自于本科学习的编译原理的，画出有限状态自动机，然后实现。ASN1 的语法定义是 ITU-T X.691 (<https://www.itu.int/rec/T-REC-X.691-202102-I/en>)。这个文档的阅读是地狱级的，所以我觉得自己的学习能力是很棒的。

工作过程：后端利用 python 分析规范文档，导出 json 格式的对象数据。前端页面最初用的用过 jquery 和纯 javascript 开发的。后来演进选择了 brython。整个做成静态页面是因为没有服务器资源，做成静态页面也好分享给其他人使用。

这个工具用到的技术栈还是挺多的，做完这个工具我熟悉并有能力使用 HTML、JS、CSS、DOM。用到的软件技巧也挺多，递归调用去计算树形前缀。最初每个消息的语法树是后端生成，然后前端载入，但这样载入页面很慢。现在语法树是点击“+”按需扩充的，都是前端代码完成的。

**项目四：这就是我简历提到的亮点 2。** <https://liuyonggang1.github.io/3GPP/msc/msc.html>

点击 demo2 按钮。

这个是画 MSC 图的，系统架构师的高频工具。

**项目五：示例：** <https://liuyonggang1.github.io/resgrid/nr/csirs1/csirs1.html>

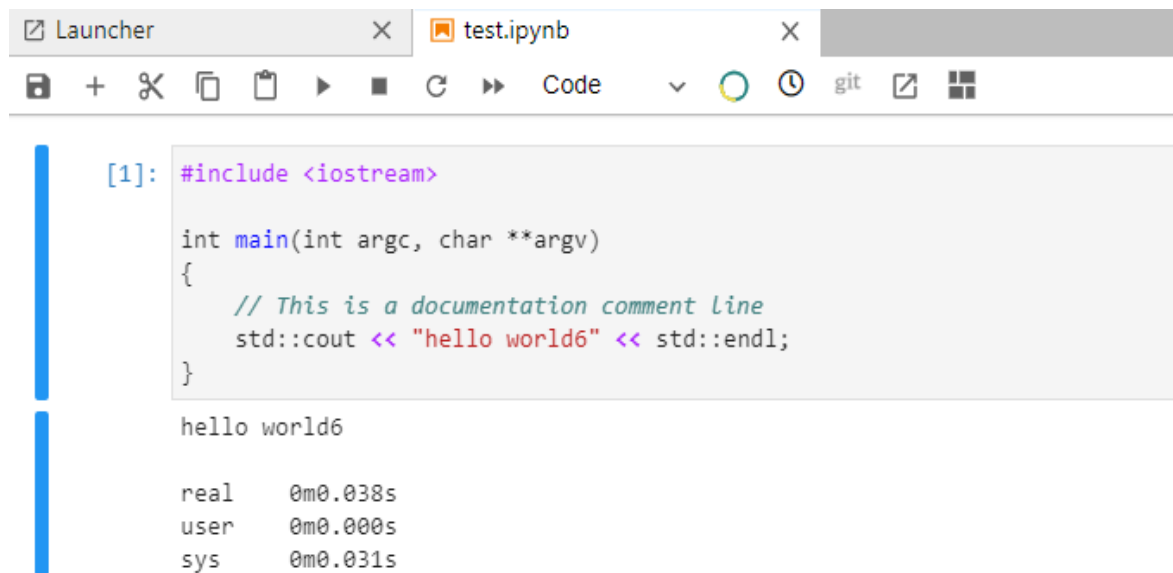
这个例子中我用 brython 把输入表格总结了自己的一个小框架。只需要关注参数定义和参数间的限制。界面由框架自动完成。

```
t = Parameter('row')
t.range = range(1, 19)
t.spec = "38.211 table 7.4.1.5.3-1"
t = Sep()
```

点击“show result”输出的表格我也总结了一个使用模式，类似 pandas 中 DataFrame 导出 HTML 文本。

**最后其它的一些杂项：**

从 2020 年起我进入 5G 仿真，这是一个大型的 C++ 项目。学习阶段我开发脚本画出所有类的继承图和协作图供自己参考。后来为了方便验证小代码段的算法，方便合作讨论，我开发了一个 Jupyter 的 C++ 的 kernel。我把生产环境的 docker image 导出了一个实例，copy 到 window10 下，创建了一个 WSL 实例。我的 kernel 就是在 WSL 下编译运行代码，再返回结果到 Jupyter。



The screenshot shows a Jupyter Notebook window with two tabs: 'Launcher' and 'test.ipynb'. The 'test.ipynb' tab is active. The interface includes a toolbar with icons for saving, adding, deleting, copying, pasting, running, and other actions. Below the toolbar, the code cell [1]: contains the following C++ code:

```
#include <iostream>

int main(int argc, char **argv)
{
    // This is a documentation comment line
    std::cout << "hello world6" << std::endl;
}
```

The output of the code cell is:

```
hello world6

real    0m0.038s
user    0m0.000s
sys     0m0.031s
```

最后的总结：我是一个持续的学习者，向往崇尚极客精神，热爱软件开发。