Go on
https://github.com/FPGA-Research-Manchester/FABulous

and follow the instructions to install FABulous. The VM image may contain already a demo project, so we start our tutorial with a fresh demo. After you changed into your FABulous folder (in your home) with **`cd FABulous`**, set the FABulous root directory:

```
export FAB_ROOT=.
python3 FABulous.py -c demo_2
```

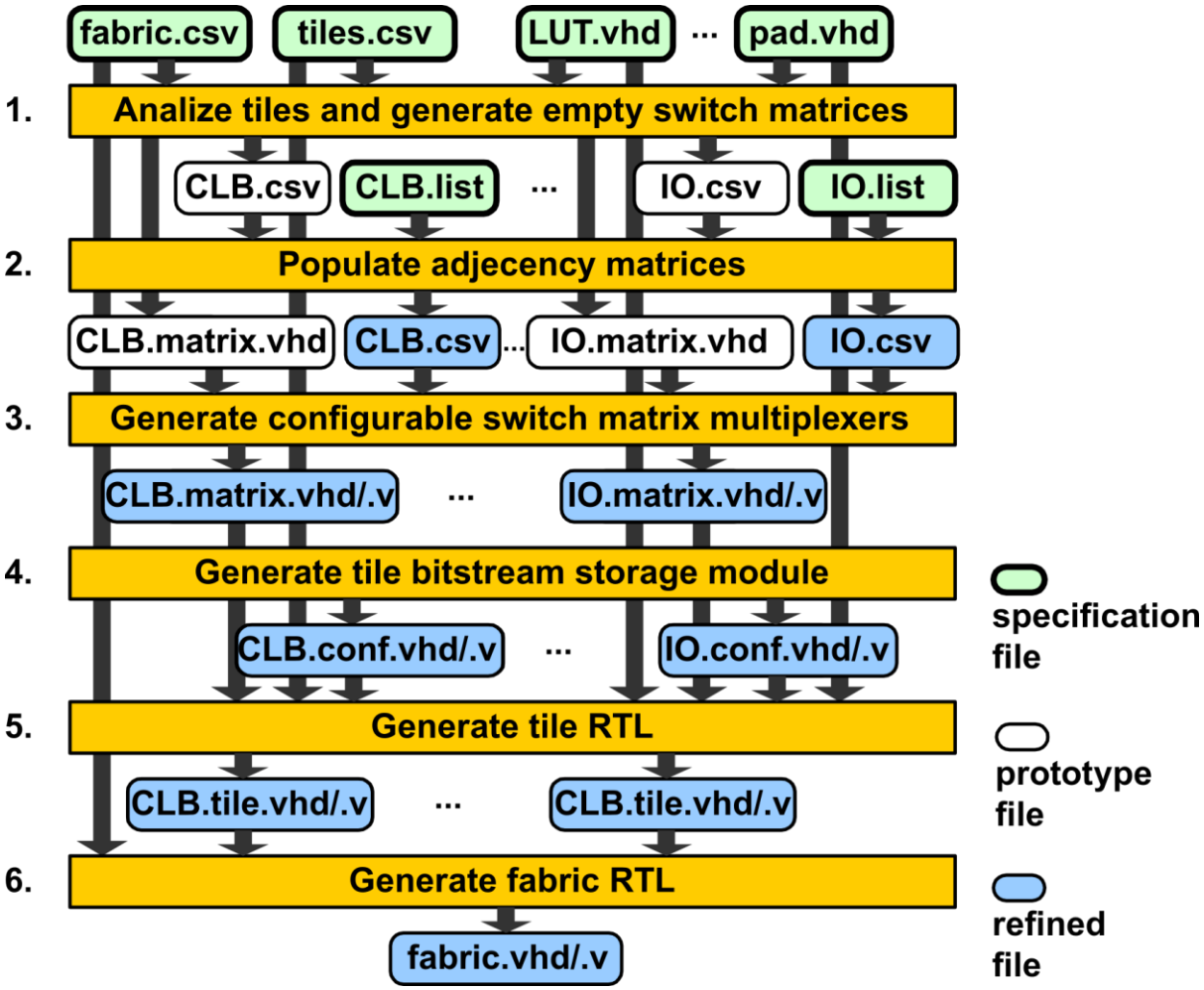
With this we can the FABulous shell

```
python3 FABulous.py demo_2
```

Type help to get some idea about the internals of the tool
Check the FABulous documentation on:

https://fabulous.readthedocs.io/en/latest/Building%20fabric.html

Before we start, have a look into the specification files provided in demo_2.



In particular, check the files for the **LUT4AB** in the **Tile** folder.
The following instruction will load the default fabric.csv fabric description and run the flow in one go:

```
load_fabric
# In the FABulous shell (python3 FABulous.py demo)
load_fabric
run_FABulous_fabric
exit
```

Check the folder again for the generated output. The final fabric is located in located in the subdirectory **Fabric**.

Lab Tasks

1. Reduce the size of the demo fabric to 8CLBs (LUT4AB) in height and 6 CLBs in width plus an extra BRAM column as in the demo (we do not use a DSP block). Or any size as long as you stay to a mod-2 hight (because of the BRAM).

2. Check out how the W_IO tiles work. They just interface external wires to the fabric using fundamentally some buffers.
   With that Knowledge, design a simple I/O tile for the top border of the chip that is not using any configuration bits. This means, we have only plain inputs or outputs (optional with flops, if you like) and that routing is carried out through the adjacent switch matrix below.

3. Create a copy of the CLB tile and name it LUT6.
   You have to copy it in fabric.csv and create a corresponding folder in Tiles. Replace the 8 LUT-4 primitives with 2 LUT-6.
   Recycle inputs and outputs from the original LUT4AB primitive.
   Recycling means that you use original LUT-4 inputs to drive your LUT-6 inputs (and the same for the outputs).

The last two exercises explain you how to integrate your own fabric into the demo baseline project. Another interesting feature related to integrating your own IP is used in RAM_IO where we export configuration bits to external pins that can then be used by an attached (user defined) block.