



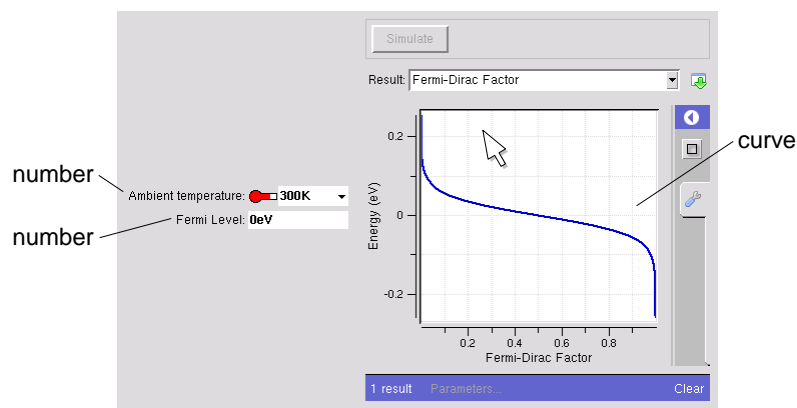
What's Under the Hood?

Michael McLennan
HUBzero® Platform for Scientific Collaboration
Purdue University

This work licensed under  See license online: [by-nc-sa/3.0](https://creativecommons.org/licenses/by-nc-sa/3.0/)

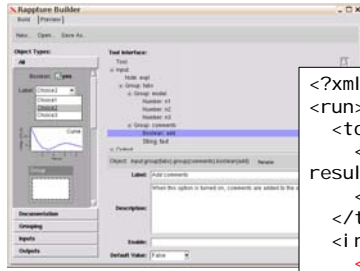


Example: Fermi function tool



How does it work?

Builder



```
<?xml version="1.0"?>
<run>
  <tool>
    <about>Press Simulate to view
    results.</about>
    <command>@tool/fermi @driver</command>
  </tool>
  <input>
    <number id="temperature">...</number>
    <number id="Ef">...</number>
  </input>
  <output>
    <curve id="F12">...</curve>
  </output>
</run>
```



skeleton program



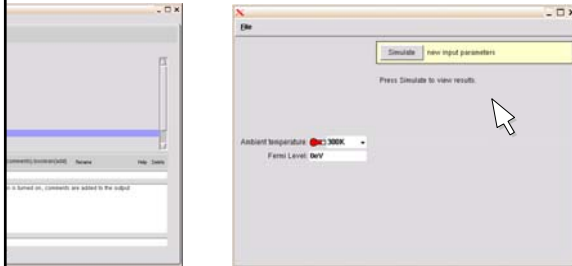
tool.xml

- Info about the tool, how to run it
- Input objects
- Output objects

Color xterm
\$ rapture

Builder

Runner



driver1827.xml



tool.xml



Running produces a driver file

```
<?xml version="1.0"?>
<run>
...
  <i nput>
    <number id="temperature">
      <uni ts>K</uni ts>
      <mi n>0K</mi n>
      <max>500K</max>
      <default t>300K</default t>

    </number>
    <number id="Ef">
      <uni ts>eV</uni ts>
      <default t>0eV</default t>

    </number>
  </i nput>
  <output>
    <curve id="f12">...</curve>
  </output>
</run>
```

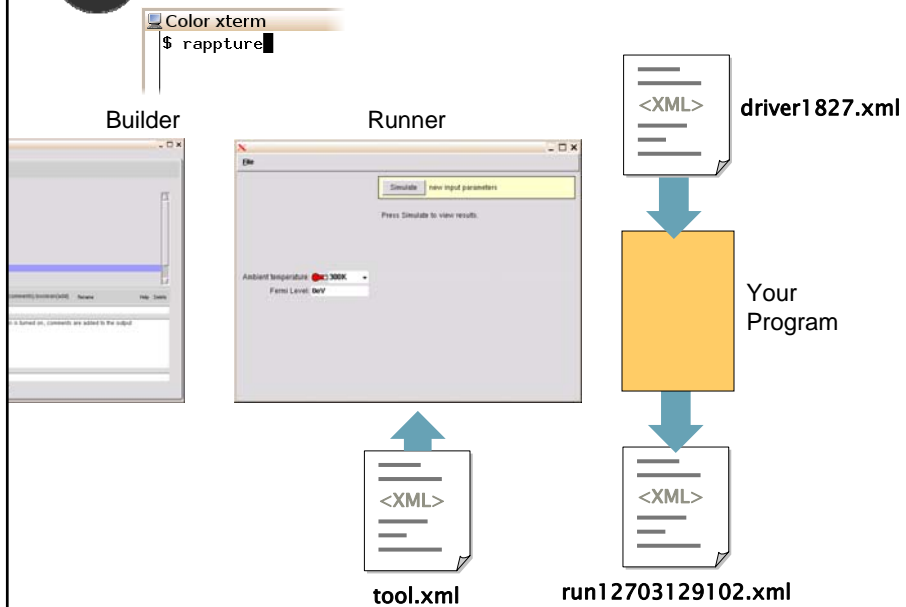
tool.xml

```
<?xml version="1.0"?>
<run>
...
  <i nput>
    <number id="temperature">
      <uni ts>K</uni ts>
      <mi n>0K</mi n>
      <max>500K</max>
      <default t>300K</default t>
      <current>77K</current>
    </number>
    <number id="Ef">
      <uni ts>eV</uni ts>
      <default t>0eV</default t>
      <current>200meV</current>
    </number>
  </i nput>
  <output>
    <curve id="f12">...</curve>
  </output>
</run>
```

driver1827.xml



Running your program





Program produces a run file

```
<?xml version="1.0"?>
<run>
...
  <i nput>
    <number id="temperature">...
      <current>77K</current>
    </number>
    <number id="Ef">...
      <current>200meV</current>
    </number>
  </i nput>
  <output>
    <curve id="f12">
      <about>...

    </curve>
  </output>
</run>
```

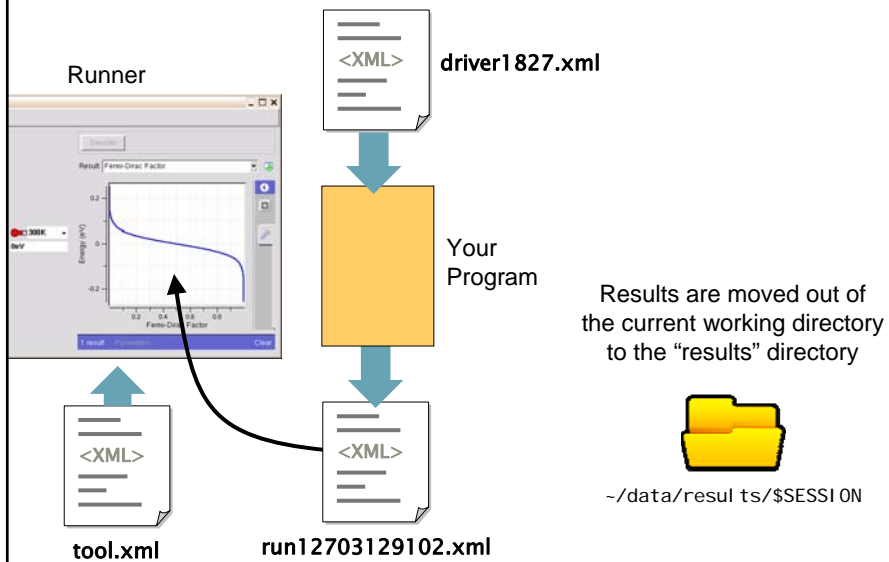
driver1827.xml

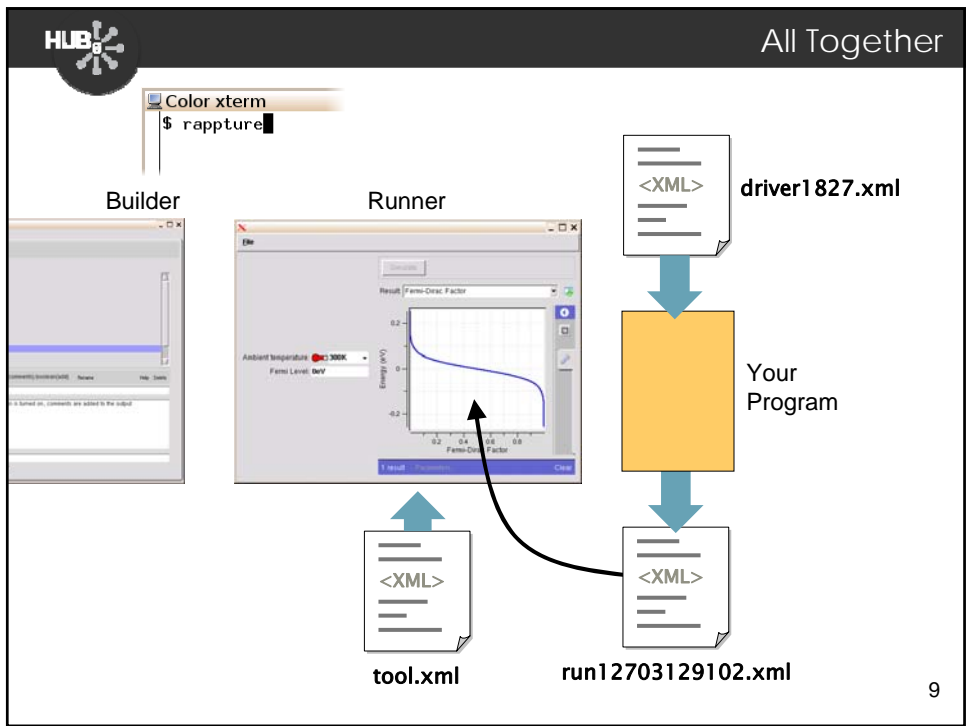
```
<?xml version="1.0"?>
<run>
...
  <i nput>
    <number id="temperature">...
      <current>77K</current>
    </number>
    <number id="Ef">...
      <current>200meV</current>
    </number>
  </i nput>
  <output>
    <curve id="f12">
      <about>...
      <component>
        <xy>0.999955 -0.25852
0.99995 -0.255935
0.999945 -0.25335...
      </curve>
    </output>
  </run>
```

run12703129102.xml



Tool Definition File





HUB Run file is a complete record of the run

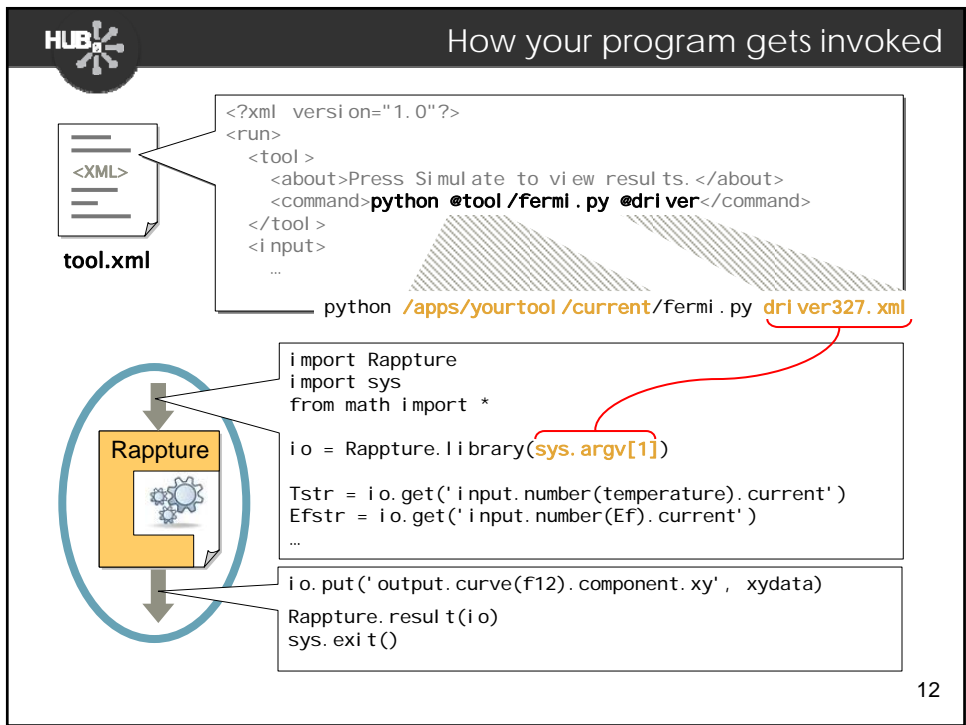
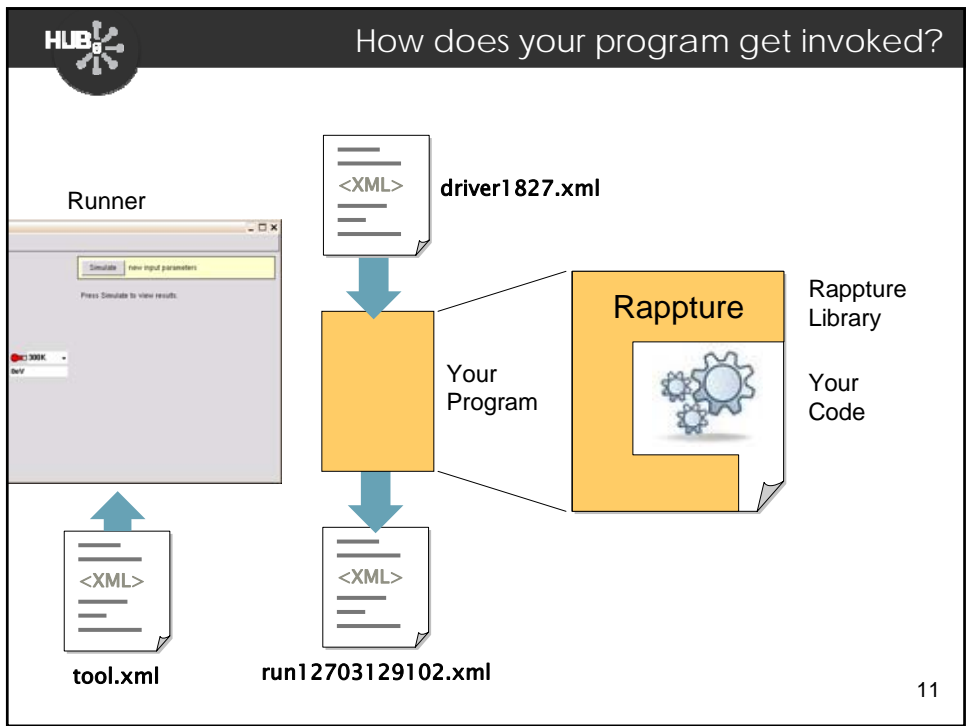
	Tool Defn	Input Vals	Output Vals
tool.xml	✓		
driver.xml	✓	✓	
run.xml	✓	✓	✓

run12703129102.xml

Color xterm
\$ rapture -load run1301535652532513.xml
reload old results

Color xterm
\$ rapture -tool run1301535652532513.xml
rerun an old tool

10



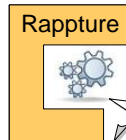


tool.xml

```
<?xml version="1.0"?>
<run>
...
<i nput>
  <number id="temperature">
    <current>300K</current>
  </number>
</i nput>
...
<output>
  <curve id="f12">
    <component>
      <xy>2.102 6.454
    ...
  ...
</output>
```

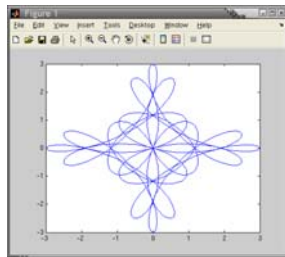
```
input
  . number(temperature)
  . current

output
  . curve(f12)
  . component
  . xy
```



Your Program

```
...
Tstr = i o.get(' i nput. number(temperature). current' )
Efstr = i o.get(' i nput. number(Ef). current' )
i o.put(' output. curve(f12). component. xy' , xydata)
...
```



Spirograph equation:

$$z(t) = e^{i2\pi n_1 t} + e^{i2\pi n_2 t} + e^{i2\pi n_3 t}$$

See theory at <http://linuxgazette.net/133/luana.html>

Where t has 1,000 points along $[0,1]$

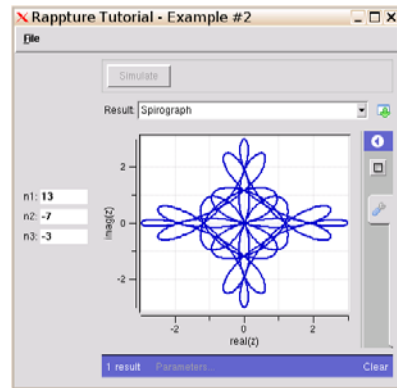
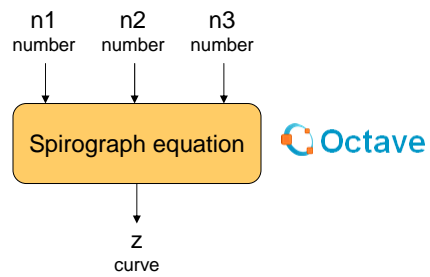
Plot:
 real(z) → x
 imag(z) → y

In MATLAB/Octave:

```
t = linspace(0, 1, 1000);
z = exp(i * 2*pi * n1*t) + exp(i * 2*pi * n2*t) + exp(i * 2*pi * n3*t);
plot(real(z), imag(z));
```



Exercise #2: Build a simple Addition tool



In MATLAB/Octave:

```
t = linspace(0, 1, 1000);  
z = exp(i * 2 * pi * n1 * t) + exp(i * 2 * pi * n2 * t) + exp(i * 2 * pi * n3 * t);  
plot(real(z), imag(z));
```