

DOL实例分析&编程

实验任务：

- 1、修改example1，使其输出3次方数，tips：修改square.c
- 2、修改example2，让3个square模块变成两个，tips：修改xml的iterator

实验过程：

- 1、了解example中各个文件的含义：看看课件就可以。
- 2、修改代码：

- 修改example1:

首先进入文件夹 dol/examples/example1/src,找到square.c文件:

```
int square_fire(DOLProcess *p) {  
    float i;  
  
    if (p->local->index < p->local->len) {  
        DOL_read((void*)PORT_IN, &i, sizeof(float), p);  
        i = i*i;  
        DOL_write((void*)PORT_OUT, &i, sizeof(float), p);  
        p->local->index++;  
    }  
}
```

可以看到其中DOL_read((void*)PORT_IN,&i,sizeof(float),p);读square的端口“PORT_IN”，将值读到i，然后进行了一个平方，然后DOL_write((void*)PORT_OUT,&i,sizeof(float),p);写square的端口“PORT_OUT”，把i写到这个端口。由此可知我们如果要输出3次方数，只需要把做平方的那一步变成做立方就可以了，即把 $i = i * i$ ，变成 $i = i * i * i$ 即可，完成后保存文件，运行example1:

```
$ cd dol/build/bin/main/  
$ ant -f runexample.xml -Dnumber=1
```

结果出错；

然后仔细检查了一下，发现build/bin/main/example1下的profile文件被锁定了，这说明它无法被修改，所以程序运行才会出错。因为其被锁定也没法被删除，所以查了下资料，然后先在同一目录下新建了一个名为profile1的文本文件，然后把profile文件的内容全都复制到了新文件，然后执行了如下命令：

```
$ sudo rm dol/build/bin/main/profile.txt -r  
$ 输入密码
```

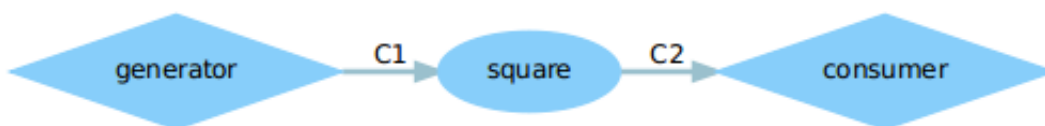
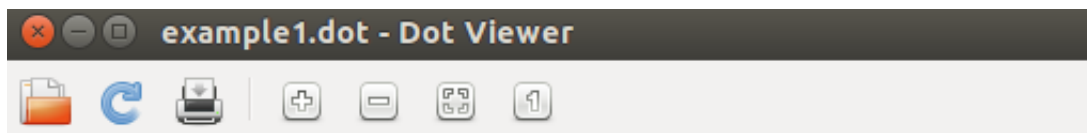
然后进入文件夹，发现被锁定的profile已经被删除，然后把profile1重命名为profile，然后用和上面一样

的指令重新运行example1，得到结果如下，可见输出的结果都是3次方数，结果正确：

```
wzh@14353308_wangzhonghua: ~/dol/build/bin/main
[concat] consumer: 0.000000
[concat] consumer: 1.000000
[concat] consumer: 8.000000
[concat] consumer: 27.000000
[concat] consumer: 64.000000
[concat] consumer: 125.000000
[concat] consumer: 216.000000
[concat] consumer: 343.000000
[concat] consumer: 512.000000
[concat] consumer: 729.000000
[concat] consumer: 1000.000000
[concat] consumer: 1331.000000
[concat] consumer: 1728.000000
[concat] consumer: 2197.000000
[concat] consumer: 2744.000000
[concat] consumer: 3375.000000
[concat] consumer: 4096.000000
[concat] consumer: 4913.000000
[concat] consumer: 5832.000000
[concat] consumer: 6859.000000

BUILD SUCCESSFUL
Total time: 5 seconds
wzh@14353308_wangzhonghua:~/dol/build/bin/main$
```

.dot文件如下：



至此，对于example1的修改任务结束。

- 修改example2:

根据提示，先找到example2.xml文件：

```
example2.xml x
<?xml version="1.0" encoding="UTF-8"?>
<processnetwork xmlns="http://www.tik.ee.ethz.ch/~shapes/schema/PROCESSNETWORK"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.tik.ee.ethz.ch/~shapes/schema/PROCESSNETWORK
http://www.tik.ee.ethz.ch/~shapes/schema/processnetwork.xsd" name="example2">

  <variable value="3" name="N"/>

  <!-- instantiate resources -->
  <process name="generator">
    <port type="output" name="10"/>
    <source type="c" location="generator.c"/>
  </process>

  <iterator variable="i" range="N">
    <process name="square">
      <append function="i"/>
      <port type="input" name="0"/>
      <port type="output" name="1"/>
      <source type="c" location="square.c"/>
    </process>
  </iterator>
```

通过之前对代码的了解可以看到example2通过迭代定义了三个平方模块，我们需要做的就是把迭代的次数从3变为2，那么就很简单了，修改如下：

```
example2.xml x
<?xml version="1.0" encoding="UTF-8"?>
<processnetwork xmlns="http://www.tik.ee.ethz.ch/~shapes/schema/PROCESSNETWORK"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.tik.ee.ethz.ch/~shapes/schema/PROCESSNETWORK
http://www.tik.ee.ethz.ch/~shapes/schema/processnetwork.xsd" name="example2">

  <variable value="2" name="N"/>
```

运行example2:

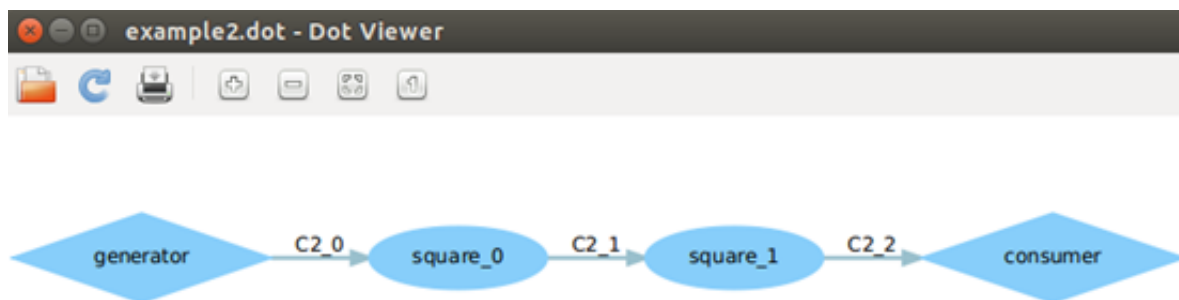
```
$ ant -f runexample.xml -Dnumber=2
```

结果如下,由于是两个平方模块，所以输出的是 i 的 4 次方数，结果正确：

```
wzh@14353308_wangzhonghua: ~/dol/build/bin/main
[concat] consumer: 0.000000
[concat] consumer: 1.000000
[concat] consumer: 16.000000
[concat] consumer: 81.000000
[concat] consumer: 256.000000
[concat] consumer: 625.000000
[concat] consumer: 1296.000000
[concat] consumer: 2401.000000
[concat] consumer: 4096.000000
[concat] consumer: 6561.000000
[concat] consumer: 10000.000000
[concat] consumer: 14641.000000
[concat] consumer: 20736.000000
[concat] consumer: 28561.000000
[concat] consumer: 38416.000000
[concat] consumer: 50625.000000
[concat] consumer: 65536.000000
[concat] consumer: 83521.000000
[concat] consumer: 104976.000000
[concat] consumer: 130321.000000

BUILD SUCCESSFUL
Total time: 14 seconds
wzh@14353308_wangzhonghua:~/dol/build/bin/main$
```

.dot文件如下:



实验感想:

本次实验比较简单, 实验过程都是根据PPT的提示进行, 没有遇到太大的问题。其中修改example1时因为profile文件被锁定而无法被修改算是比较重要的一个问题, 但是通过查资料, 很快就解决了。