Application

To create an application for DSX-VM we need to:

- write the code of different task and describe the task in a python file
- describe the TCG (task and communication graph) of the application

Task writing

To define a task you need to write down a least two files:

- a *.c file containing the code of the task : the task is writed using the SRL API
  - example :

```
/* those two first header must be declared */
#include <srl.h> // same for all task and contain the definition of the SRL API fun
#include "hello_proto.h" // containing specific definition for the task

FUNC(hello_func)
{
    while(1)
    {
        srl_log_printf(NONE, "Hello world\n");
    }
}
```

- a *.tsk file containing a python description (metadata) of the the task. The task is described by using TaskModel? python class:

```
TaskModel( name, ports, impls, uses)
```

  - name : a string, describing the name of the application
  - ports : a dictionary, describing the ports of communication and their names. Different port exist:
    - MwmrInput : a mwmr input port
    - MwmrOutput : a mwmr output port
    - BarrierPort : a barrier port
    - LockPort : a lock port
    - MemspacePort : a memspace port
  - impls : a list, describing the available implementations supported by the task
    - swtask
  - uses : a list of string, describing the used resources by the task (example: 'tty')

TCG

The application is described by a TCG (task an communication graph). The TCG describe the flow of the application. It describe

- the interconnection
- the mean of communication (generally mwmr channel)

beetween task.

Example : This is a TCG of an application, with three task communicating through mwmrs channels.