

What is a promise

Promise is an executable doing some arbitrary work, then exiting with exit code 0 ("it works") or greater ("it doesn't work"). Promises are generated during instantiation in `$instance_home/etc/promise`, then all promises in this directory are run by slapgrid to know if instance is working or not.

The most simple example of promise is "check_if_port_listening" (http://git.erp5.org/gitweb/slapos.git/tree/HEAD:/slapos/recipe/check_port_listening) trying to open a socket to an ip/port. If it works, it exits with exit code 0, and slapgrid knows that the instance is working. If socket can't be created, it exits with another exit code, and slapgrid reports it to the SlapOS Master.

The code of the promise itself is in the template directory, while the recipe used to "install" the promise into the instance is in `__init.py__`.

All software and stacks such as NBD (<http://git.erp5.org/gitweb/slapos.git/blob/HEAD:/software/kvm/instance-nbd.cfg?js=1#l37>) should use the promise system to define as precisely as possible if instance is working or not.

How to add an existing promise to a Software Release

In your instance profile, add a part looking like this:

```
[frontend-promise]
recipe = slapos.cookbook:check_url_available
path = ${directory:promises}/apache_promise
url = ${publish-kvm-frontend-connection-information:url}
dash_path = ${dash:location}/bin/dash
curl_path = ${curl:location}/bin/curl
```

Don't forget to add it in the parts list.

This will generate an executable in `${directory:promises}/apache_promise` that, when run, will try to connect to the url. Here, we need to specify path for dash and curl executables. Of course, argument values may differ from this example.

You can add as many promises as you need.

How to add a new promise to a Software Release

- Write your recipe responsible of generating the executable
- Add the part calling it, like previous section.