static void _f_do_barnacle_install_properties(GObjectClass *gobject_class)
{
    GParamSpec *pspec;
    /* Party code attribute */
    pspec = g_param_spec_uint64
        (F_DO_BARNACLE_CODE,
         "Barnacle code.",
         "Barnacle code",
         0,
         G_MAXUINT64,
         G_MAXUINT64 /* default
         value */,
         G_PARAM_READABLE |
         G_PARAM_WRITABLE |
         G_PARAM_PRIVATE);
    g_object_class_install_property (gobject_class,
        F_DO_BARNACLE_PROP_CODE,
        Adián Pérez de Castro
        aperez@igalia.com
        Shaft
        Shell Advanced
        Functional Tester
Outline

- What?
- Why?
- How?
What?
What? · The “ins”

- Shell scripting
- Simple
- Declarative
- Unit testing
- Funky name
What? · The “outs”

- Bash only
- Simple
- Still have to write tests...
- Not a silver bullet
Why?
Why? · Usual suspects

- Almost nobody wants to write tests
  - Let's make it easier
- It takes precious time to write tests
  - Let's make it easier
Why? · The Debian connection

<table>
<thead>
<tr>
<th>Item</th>
<th>Lines</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial RAM disk</td>
<td>1.200</td>
</tr>
<tr>
<td>Init scripts</td>
<td>3.000</td>
</tr>
<tr>
<td>Package scripts (~200 base packages)</td>
<td>17.300</td>
</tr>
<tr>
<td>Entire file system</td>
<td>88.000</td>
</tr>
</tbody>
</table>
Why? · The Debian connection

- Package scripts are tested manually
  - Some of them mission-critical
- Init scripts
- Debian installer?
Why? · Existing tools

- echo
- set -x
- set -u
- set -v
- Bashdb
- shUnit2
Why? · Existing tools

- echo
- set -x
- set -u
- set -v
- Bashdb
- shUnit2
Why? · Existing tools

- echo
- set -x
- set -u
- set -v

- Bashdb
- shUnit2
How?
How? · The stack

Shaft

Bashdb

shUnit2

Bash
How? · Overview

- Put marks in the code
  - Prefixed with `#+`
- Write some proof files
  - Augmented bashdb command script
- Write a runner script
How? · Marks in script

age=45
#+ test01 test03
if [[ ${age} -gt 0 ]] ; then
  #+ test02
  echo "Age: ${age}"
fi
How? · Runner script

```bash
#!/bin/bash
SHAFT_SOURCE="example-script"
SHAFT_PROOFDIR="."
source shaft
```
breaks () {
    at 1
}
start
gt 0 echo ${age}
breaks () {
    at 1
}
start
= entered-if echo entered-if
How? · test03.proof

breaks () {
    at 1
    at 2
}

start

gt 0 echo ${age}
continue

= entered-if echo entered-if

DebConf
Cáceres - Spain
How? · Running

~/shaft/examples $ ./example-runner
#
# Performing tests
#
test01-age-value
test02-enter-loop
test03-both

#
# Test report
#
tests passed: 4 100%
tests failed: 0 0%
tests skipped: 0 0%
tests total: 4 100%
Wrap up
Wrap up

- Somewhat hackish...
- ...but already saved a good deal of time
- Maybe limited range of application...
- ...but the name is catchy!
Final Remarks

- Repository: http://gitorious.org/shaft
- Getting in touch: shaft@igalia.com
- Thanks for attending!