~~Make~~ Rebar Great Again

•••

March 10, 2016

Tristan Sloughter
@t_sloughter
GOFACTORY

Talk objectives:

The whys and the hows.

Donal Morrissey

3.0.0 Released?!?
Now github.com/erlang/rebar3 ??!

Adoption

heroku

PROJECTFIFO

GOFACTORY

CHEF™

erlio

adgear

BARREL

AdRoll

DIPL. PHYS.
PEER STRITZINGER GMBH

ASKUITY

sqor
SPORTS

HELIUM

Erlang

Why Rebar3

Repeatability

- Deterministic dependency resolution
- Lock dependencies

Packages

- Hex.pm:
  - Central store for BEAM language packages

Dependency Pollution

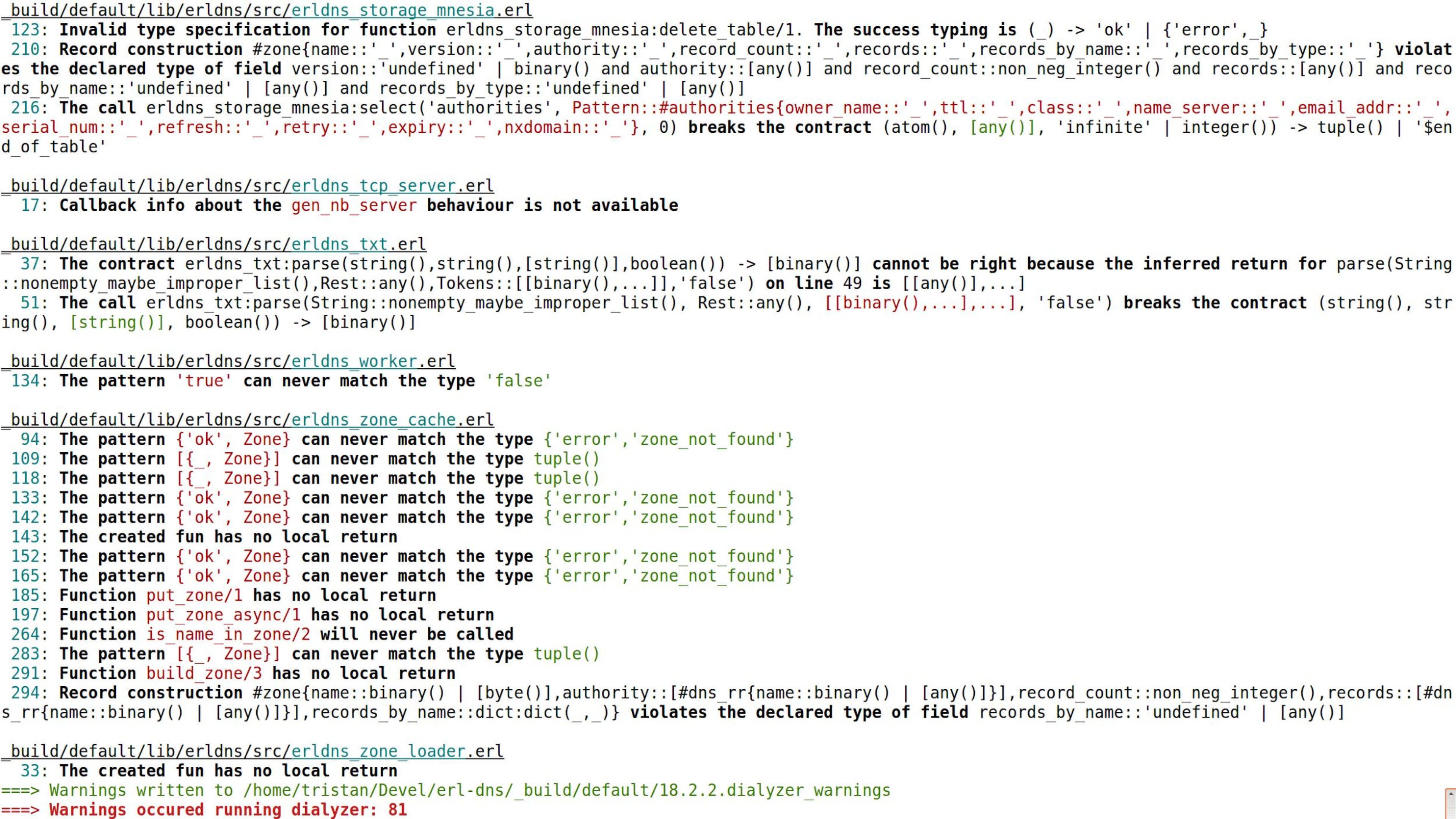
- Test and docs profiles for deps:
  - meck, edown, etc

Broken Features

- update-deps
- recursive commands

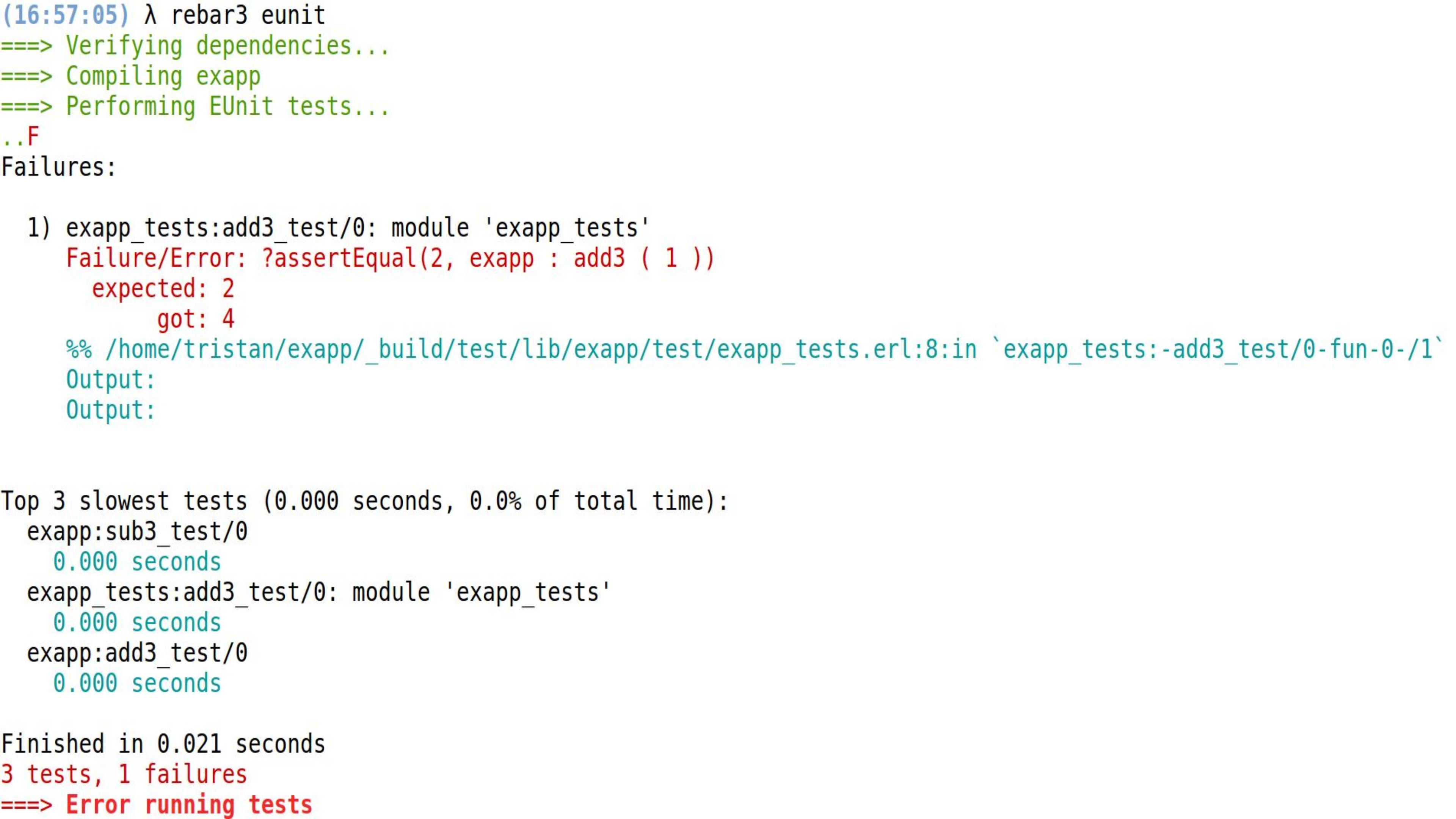
What’s New?

CHICAGO
DIALYZER:NOW IN
GLORIOUS TECHNICOLOR



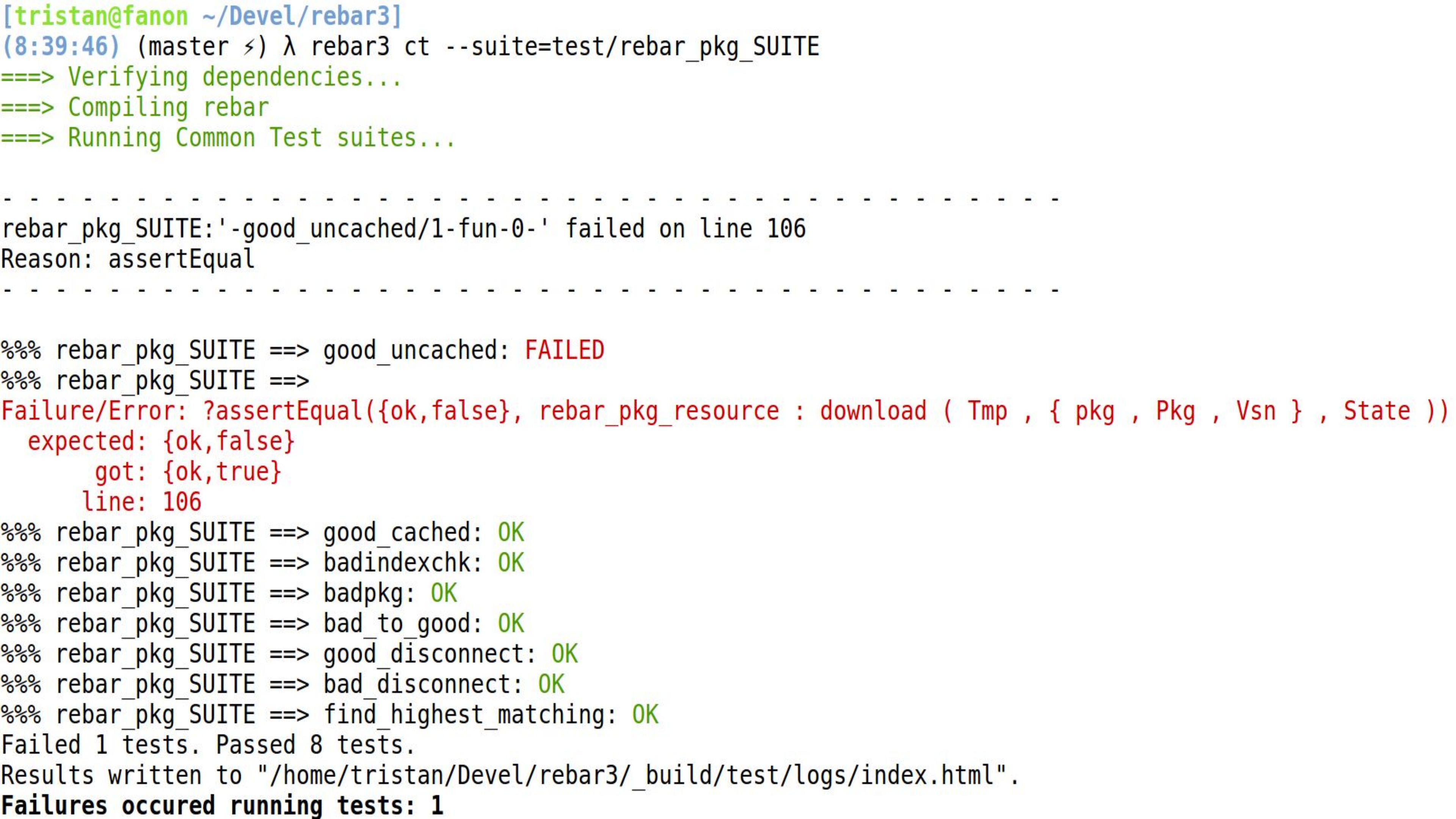
```
_build/default/lib/erldns/src/erldns_storage_mnesia.erl
 123: Invalid type specification for function erldns_storage_mnesia:delete_table/1. The success typing is (_) -> 'ok' | {'error',_}
 210: Record construction #zone{name::'_',version::'_',authority::'_',record_count::'_',records::'_',records_by_name::'_',records_by_type::'_'} violat
es the declared type of field version::'undefined' | binary() and authority::[any()] and record_count::non_neg_integer() and records::[any()] and reco
rds_by_name::'undefined' | [any()] and records_by_type::'undefined' | [any()]
 216: The call erldns_storage_mnesia:select('authorities', Pattern::#authorities{owner_name::'_',ttl::'_',class::'_',name_server::'_',email_addr::'_',
serial_num::'_',refresh::'_',retry::'_',expiry::'_',nxdomain::'_'}, 0) breaks the contract (atom(), [any()], 'infinite' | integer()) -> tuple() | '$en
d_of_table'

_build/default/lib/erldns/src/erldns_tcp_server.erl
  17: Callback info about the gen_nb_server behaviour is not available

_build/default/lib/erldns/src/erldns_txt.erl
  37: The contract erldns_txt:parse(string(),string(),[string()],boolean()) -> [binary()] cannot be right because the inferred return for parse(String
::nonempty_maybe_improper_list(),Rest::any(),Tokens::[[binary(),...]],'false') on line 49 is [[any()],...]
  51: The call erldns_txt:parse(String::nonempty_maybe_improper_list(), Rest::any(), [[binary(),...],...], 'false') breaks the contract (string(), str
ing(), [string()], boolean()) -> [binary()]

_build/default/lib/erldns/src/erldns_worker.erl
 134: The pattern 'true' can never match the type 'false'

_build/default/lib/erldns/src/erldns_zone_cache.erl
  94: The pattern {'ok', Zone} can never match the type {'error','zone_not_found'}
 109: The pattern [{_, Zone}] can never match the type tuple()
 118: The pattern [{_, Zone}] can never match the type tuple()
 133: The pattern {'ok', Zone} can never match the type {'error','zone_not_found'}
 142: The pattern {'ok', Zone} can never match the type {'error','zone_not_found'}
 143: The created fun has no local return
 152: The pattern {'ok', Zone} can never match the type {'error','zone_not_found'}
 165: The pattern {'ok', Zone} can never match the type {'error','zone_not_found'}
 185: Function put_zone/1 has no local return
 197: Function put_zone_async/1 has no local return
 264: Function is_name_in_zone/2 will never be called
 283: The pattern [{_, Zone}] can never match the type tuple()
 291: Function build_zone/3 has no local return
 294: Record construction #zone{name::binary() | [byte()],authority::[#dns_rr{name::binary() | [any()]}],record_count::non_neg_integer(),records::[#dn
s_rr{name::binary() | [any()]}],records_by_name::dict:dict(_,_)} violates the declared type of field records_by_name::'undefined' | [any()]

_build/default/lib/erldns/src/erldns_zone_loader.erl
  33: The created fun has no local return
===> Warnings written to /home/tristan/Devel/erl-dns/_build/default/18.2.2.dialyzer_warnings
===> Warnings occured running dialyzer: 81
```

EUnit Too!



```
(16:57:05) λ rebar3 eunit
===> Verifying dependencies...
===> Compiling exapp
===> Performing EUnit tests...
..F
Failures:

  1) exapp_tests:add3_test/0: module 'exapp_tests'
     Failure/Error: ?assertEqual(2, exapp : add3 ( 1 ))
       expected: 2
            got: 4
     %% /home/tristan/exapp/_build/test/lib/exapp/test/exapp_tests.erl:8:in `exapp_tests:-add3_test/0-fun-0-/1`
     Output:
     Output:


Top 3 slowest tests (0.000 seconds, 0.0% of total time):
  exapp:sub3_test/0
    0.000 seconds
  exapp_tests:add3_test/0: module 'exapp_tests'
    0.000 seconds
  exapp:add3_test/0
    0.000 seconds

Finished in 0.021 seconds
3 tests, 1 failures
===> Error running tests
```

Common Test Three!



```
[tristan@fanon ~/Devel/rebar3]
(8:39:46) (master ⚡) λ rebar3 ct --suite=test/rebar_pkg_SUITE
===> Verifying dependencies...
===> Compiling rebar
===> Running Common Test suites...

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
rebar_pkg_SUITE:'-good_uncached/1-fun-0-' failed on line 106
Reason: assertEqual
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%%% rebar_pkg_SUITE ==> good_uncached: FAILED
%%% rebar_pkg_SUITE ==>
Failure/Error: ?assertEqual({ok,false}, rebar_pkg_resource : download ( Tmp , { pkg , Pkg , Vsn } , State ))
  expected: {ok,false}
       got: {ok,true}
      line: 106
%%% rebar_pkg_SUITE ==> good_cached: OK
%%% rebar_pkg_SUITE ==> badindexchk: OK
%%% rebar_pkg_SUITE ==> badpkg: OK
%%% rebar_pkg_SUITE ==> bad_to_good: OK
%%% rebar_pkg_SUITE ==> good_disconnect: OK
%%% rebar_pkg_SUITE ==> bad_disconnect: OK
%%% rebar_pkg_SUITE ==> find_highest_matching: OK
Failed 1 tests. Passed 8 tests.
Results written to "/home/tristan/Devel/rebar3/_build/test/logs/index.html".
Failures occured running tests: 1
```

New Plugins!

C Compiler

Features:

- Same configuration as rebar2
- Hex.pm package available
- Hook in to compile provider

Cuttlefish

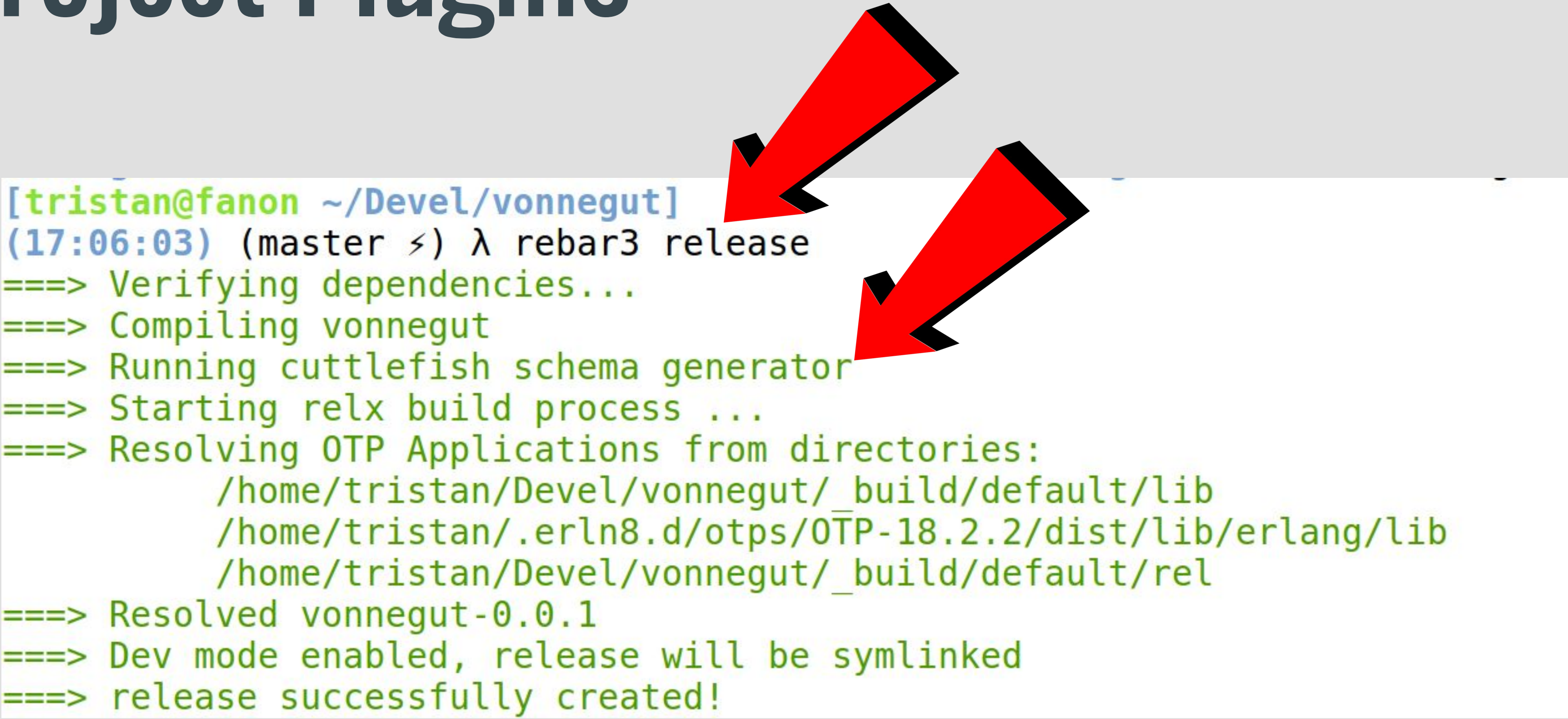
Features:

- Auto-schema discovery
- Cuttlefish enabled start script
- Overrides release and tar

New Type of Plugin:

- `{project_plugins, [rebar3_cuttlefish]}.`

Project Plugins



```
[tristan@fanon ~/Devel/vonnegut]
(17:06:03) (master ⚡) λ rebar3 release
===> Verifying dependencies...
===> Compiling vonnegut
===> Running cuttlefish schema generator
===> Starting relx build process ...
===> Resolving OTP Applications from directories:
          /home/tristan/Devel/vonnegut/_build/default/lib
          /home/tristan/.erln8.d/otps/OTP-18.2.2/dist/lib/erlang/lib
          /home/tristan/Devel/vonnegut/_build/default/rel
===> Resolved vonnegut-0.0.1
===> Dev mode enabled, release will be symlinked
===> release successfully created!
```

Many More New Plugins

- Quickcheck (rebar3_eqc)
  - Failed! (╯°□°)╯ ︵ ┻━┻ After 1 tests.
- PropEr (rebar3_proper)
- Appup (rebar3_appup_plugin)
- Protocol Buffers (rebar3_gpb_plugin)
- Relflow (relfow)
- Auto (rebar3_auto)
- Autotest (rebar3_auto_test)

Moving Your Project to Rebar3

Directory Structure

- Remove `./ebin/`
- Remove `./deps/`
- Remove `.rebar/`
- Add `_build/` to `.gitignore` or`.hgignore`

Move Your Deps

Dependencies unrelated to building should be moved to profiles

```
{deps, []}.

{profiles, [{test, [{deps, [meck]}]},
            {docs, [{deps, [edown]}]}]}.
```

Directory Structure cont.

- You’ll find apps’ output under **`_build/default/lib/<app>/ebin/`**
- No need for **`sub_dirs`** or **`lib_dirs`** in **`rebar.config`**
- Apps under **`./apps/`** or **`./lib/`** automatically detected like **`./src/`**

No Editing Dependencies’ Source Files!!

- Dependencies are compiled once
- Any change will not be compiled
- You break repeatability!
- (oh, and no raw deps)

_checkouts to the Rescue

- `ln -s ../someapp _checkouts/someapp`
- Any app under `_checkouts/` is treated like a project app
- Entry of the dep is removed from the lock file

Eunit and CT Changes

- Built under test profile, **`_build/test/`**
- Test suites are compiled with the relevant project options (like **`erl_opts`**)
- Comprehensive coverage across both tools
- No more hidden **`.eunit/`** (**`_build/test/`**)
- Eunit cli and config opts for running tests in modules, applications, suites or dir

Review .app.src File

- `applications` list used by `relx`
- No circular dependencies
- Can have hex.pm package information

Add Plugin Hooks

C Hook

(example: eleveldb)

```
{plugins, [pc]}.
{artifact, "priv/eleveldb.so"}.
{provider_hooks,
 [{pre, [
   {compile, {pc, compiler}}]}
 ]}.
```

Use Overrides

(example: eleveldb as dep)

```
{overrides,
[{override, eleveldb, [
  {artifacts, ["priv/eleveldb.so"]},
  {pre_hooks, [
   {compile,"c_src/build_deps.sh get-deps"},
   {compile,"c_src/build_deps.sh"}
  ]},
  {post_hooks, [
   {clean, "c_src/build_deps.sh clean"}
  ]},
  {plugins, [pc]},
  {provider_hooks,
   [{post,
    [{compile, {pc, compile}},
     {clean, {pc, clean}}]}]}]}
 ]}.
```

Convert to Relx

- No more reltool
- Simpler config
- Reasonable defaults
- Easy to use

reltool to relx

```
{sys,[
  {lib_dirs, ["../deps", "../apps"]},
  {rel, "relname", "0.1.0",[
    kernel,
    stdlib,
    appname,
    ...
    sasl
  ]},
  {rel, "start_clean", "", [
    kernel,
    stdlib]},
  {boot_rel, "relname"},
  {profile, embedded},
  {excl_archive_filters, [".*"]},
  {app, sasl, [{incl_cond, include}]},
  {app, cuttlefish, [{incl_cond, include}]},
  {app, lager, [{incl_cond, include}]},
  {app, appname, [{incl_cond, include}]}
 ]}.
{target_dir, "relname"}.
{overlay_vars, "vars.config"}.
{overlay, [
 {copy, "../deps/cuttlefish/cuttlefish", "{{erts_vsn}}/bin/cuttlefish"},
 {copy, "../deps/node_package/priv/base/nodetool", "{{erts_vsn}}
/bin/nodetool"},
 {template, "../deps/node_package/priv/base/runner","bin/relname"},
 {template, "files/relname.schema", "lib/00-relname.schema"},
 {template, "../deps/cuttlefish/priv/erlang_vm.schema", "lib/11-
erlang_vm.schema"},
]}.
```

```
{relx, [{release, {relname, "0.1.0"},
          [appname,
           sasl]},

        {dev_mode, true},
        {include_erts, false},

        {overlay_vars, "config/vars.config"}
        ]}.
```

Publish to Hex.pm

- `~/.config/rebar3/rebar.config`:
    - `{plugins, [rebar3_hex]}`.
- `rebar.config`:
    - `{project_plugins, [rebar3_hex]}.`
- `rebar3 hex publish` or `cut`
- https://hex.pm/docs/rebar3_publish

Many Thanks!

- Fred Hebert
- Alisdair Sullivan
- James Fish
- Heinz Gies
- IEUG and Ericsson
- Many, many more...

Questions?