



# PACKAGING ERLANG APPLICATIONS

EUC 2013 - Jared Morrow



# FULLBOKAT



# Who Am I?

- Software Engineer @ Basho
- [jared@basho.com](mailto:jared@basho.com)
- [@\\_jared](#)
- [github/jaredmorrow](#)
- Freenode: Jared\_M





# Shipping Packages



# Topics


- Why is packaging your app important?
- Why use a tool (is it really so hard)?
- What is node\_package?
- How node\_package works (hint, "magic")?
- How to package your app walkthrough with Stanchion






# WHY ARE PACKAGES IMPORTANT?



- 
- Most users download packages over source
  - Packages are the first impression for your users
  - Packages make install/upgrade/automation easier



- 
- Most users download packages over source
  - Packages are the first impression for your users
  - Packages make install/upgrade/automation easier






# User Downloads

```
riak-1.2.1.tar.gz 16  
  
riak-1.2.1-1.el6.x86_64.rpm 5  
riak-1.2.1-1.src.rpm 2  
riak_1.2.1-1_amd64.deb 80  
riak-1.2.1-osx-x86_64.tar.gz 11  
riak_1.2.1-1_i386.deb 2  
riak-1.2.1-1.el5.x86_64.rpm 169
```

- First day of Riak 1.2.1 Release
- 16 source downloads
- 169 downloads just for the RHEL 5 RPM




- 
- Most users download packages over source
  - Packages are the first impression for your users
  - Packages make install/upgrade/automation easier



# First Impressions

- Package Install is the first impression
- [Bad | Good] Quality in packaging may imply [Bad | Good] Engineering
- Users might not get past bad installs



- 
- Most users download packages over source
  - Packages are the first impression for your users
  - Packages make install/upgrade/automation easier



# Easiest Path

- Package Erlang with Application
- Users are not concerned with large packages if they work
- Chef/Puppet/Ansible become vastly easier

```
case node['riak_cs']['package']['type']
when "binary"
  case node['platform']
  when "ubuntu"
    machines = {"x86_64" => "amd64"}
    base_uri = "#{base_uri}#{node['package']}"
    package_file = "#{base_filename}
  when "debian"
    machines = {"x86_64" => "amd64"}
    base_uri = "#{base_uri}#{node['package']}"
    package_file = "#{base_filename}
  when "redhat", "centos", "scientific"
    machines = {"x86_64" => "x86_64"}
    base_uri = "#{base_uri}rhel/#{node['package']}"
    package_file = "#{base_filename}
  when "fedora"
    machines = {"x86_64" => "x86_64"}
    base_uri = "#{base_uri}#{node['package']}"
    package_file = "#{base_filename}
  end
end
```





# WHY **YOU** SHOULD DO THE PACKAGING

If you don't, others will.





WHO OWNS MY QUALITY?



**YOU**





```
1 The following NEW packages will be installed:
2   erlang erlang-appmon erlang-asn1 erlang-base erlang-common-test erlang-corba
3 erlang-crypto erlang-debugger erlang-dev erlang-dialyzer erlang-diameter
4   erlang-docbuilder erlang-edoc erlang-erl-docgen erlang-et erlang-eunit erlang-
5 examples erlang-gs erlang-ic erlang-ic-java erlang-inets erlang-inviso
6   erlang-jinterface erlang-megaco erlang-mnesia erlang-mode erlang-observer
7 erlang-odbc erlang-os-mon erlang-parsetools erlang-percept erlang-pman
8   erlang-public-key erlang-reltool erlang-runtime-tools erlang-snmp erlang-src
9 erlang-ssh erlang-ssl erlang-syntax-tools erlang-test-server erlang-toolbar
10  erlang-tools erlang-tv erlang-typer erlang-webtool erlang-xmerl libltdl7
11 libodbc1 libsctp1 lksctp-tools
12 0 upgraded, 51 newly installed, 0 to remove and 5 not upgraded.
13 Need to get 43.8 MB of archives.
14 After this operation, 112 MB of additional disk space will be used.
15 Do you want to continue [Y/n]? n
16 Abort.
17
18
```



## Download Erlang OTP

Erlang Solutions has built and tested Erlang/OTP extensively using the full OTP test suite. The following is a list of ready-to-go packages of the latest release download (R16B), for various operating systems:

Version

Operating System

Version	OS	OS Vsn	Pkg	MD5	Sources	Tests
R16B 64bit	Ubuntu	13.04				
R16B 64bit	Ubuntu	12.10				






















# Much Better





?? # \$ % @ #

"I don't even...."

Name	Approved?
Riak 	Yes
erlang-riak_core 	Yes
erlang-lager 	Yes
erlang-webmachine 	Yes
erlang-mochiweb 	Yes
rebar 	Yes
erlang-getopt 	Yes
erlang-protobuffs 	Yes
erlang-lfe 	Yes
erlang-neotoma 	Yes
erlang-mustache 	Yes
erlang-riak_sysmon 	Yes
erlang-poolboy 	Yes
erlang-bitcask 	Yes
erlang-ebloom 	Yes
erlang-riak_kv 	Yes
erlang-merge_index 	Yes
erlang-sexp 	Yes
erlang-edown 	Yes
erlang-basho_metrics 	Yes
erlang-eper 	Yes



# Dependencies

```
# Riak 1.3.1 `rebar list-deps`
basho_stats TAG 1.0.3
bear TAG 0.1.2
bitcask TAG 1.6.1
cluster_info TAG 1.2.3
ebloom TAG 1.1.2
eleveldb TAG 1.3.0
eper TAG 3280b736
erlang_js TAG 1.2.2
erlydtl TAG c18f2a00
folsom TAG 0.7.3p1
getopt TAG v0.4.3
lager TAG 1.2.2
lager_syslog TAG 1.2.2
meck TAG 0.7.2-2-gd6230ae
merge_index TAG 1.3.0
mochiweb TAG 1.5.1p3
poolboy TAG 0.8.1-14-g0e15b5d
...many more
```

- What happens if someone upgrades just 'erlang-lager'?
- Lager 2.x  $\neq$  Lager 1.x
- Our choices should not affect other Erlang applications
- How do you upgrade from Riak 1.2 to 1.4?



~~WHY~~ **YOU** SHOULD DO THE  
PACKAGING!



# HOW TO PACKAGE





Made From Scratch!







Use A Tool





# WHY MAKING CUSTOM PACKAGES IS ANNOYING IN THREE SLIDES

Not enough time for 3,000 slides



# **Debian New Maintainers' Guide**

Josip Rodin and Osamu Aoki

June 5, 2013

60 Pages



# **The pkgsrc guide**

## **Documentation on the NetBSD packages system**

**(2007/09/18)**

171 Pages



# Maximum RPM

## Taking the RPM Package Manager to the Limit

Edward C. Bailey, Red Hat, Inc.

%ghost description: Paul Nasrat

Start of updates, epoch, rpmbuild, etc: Matthias Saou

Various typo fixes, %check section, documentation on --recompil: Ville Skyttä

# 409 Pages





Use A Tool



# WHAT IS NODE\_PACKAGE





# A HISTORY

“In the beginning...”






# Addition Of Packaged Platforms

- Riak 0.14 (Dec. 2010) **5** Builds
- Riak 1.0 (Sep. 2011) **8** Builds <- **Add OSX**
- Riak 1.1 (Feb. 2012) **9** Builds
- Riak 1.2 (Aug. 2012) **11** Builds <- **Add SmartOS & FreeBSD**
- Riak 1.3 (Feb. 2013) **14** Builds <- **Add SmartOS 1.8**
- Riak 1.4 (TBD) **14** Builds <- **Dropped 32bit support**




PUBLIC  basho / node\_package Pull Request

[Code](#) [Network](#) [Pull Requests](#) 1 [Issues](#) 12

tag: 0.1.0 [Files](#) [Commits](#) [Branches](#) 4

**Initial commit**  
master 1.2.2 ... 0.1.0

 **dizzyd** authored 2 years ago 0 parents

Showing 23 changed files with 977 additions and 0 deletions.

# Dizzy To The Rescue

April 2011



Dec 22, 2011



**Bring debian and rpm builds to proper spec and refactor basho-isms out**

jaredmorrow authored a year ago

Apr 20, 2011



**Updating init.script for RPM builds to be generic**

dizzyd authored 2 years ago



basho / riak\_cs

Pull Request Unwatch Star

Code Network Pull Requests 7 Issues 73 Wiki Graphs

tree: 7bc0fd8ed2 Files Commits Branches 98

### add node\_package dep

0.0.0

argv0 authored 2 years ago 1 parent 6352011 commit 7bc0fd8ed2a40e096d87

Showing 1 changed file with 1 addition and 0 deletions.

rebar.config

```
@@ -10,6 +10,7 @@
...
10 10
11 11
12 12 {deps, [
13 + {node_package, ".*", {git, "git://github.com/basho/node_package", "HEAD"}},
```

# RiakCS Was Born!


October 2011










branch: 1.2 ▾ Files Commits Branches 124

riak / package / +

Merge pull request #230 from basho/jsm-pkg-fixes ...

 jaredmorrow authored 7 months ago lat

..

 deb	7 months ago	Merge pull request #230 from basho/jsm-pkg-fixes [jaredmorrow]
 freebsd	8 months ago	Add lager vars.config settings to packages [jaredmorrow]
 osx	a year ago	Determine package name based on dependency changes [jaredmorrow]
 rpm	8 months ago	Add lager vars.config settings to packages [jaredmorrow]
 smartos	8 months ago	Add lager vars.config settings to packages [jaredmorrow]
 solaris	a year ago	Sync up vars.config amongst packages and clean up descriptions [jaredmorrow]
 Makefile	8 months ago	Embeds distro name into version tag. [jsmartin]

# Riak 1.2.0

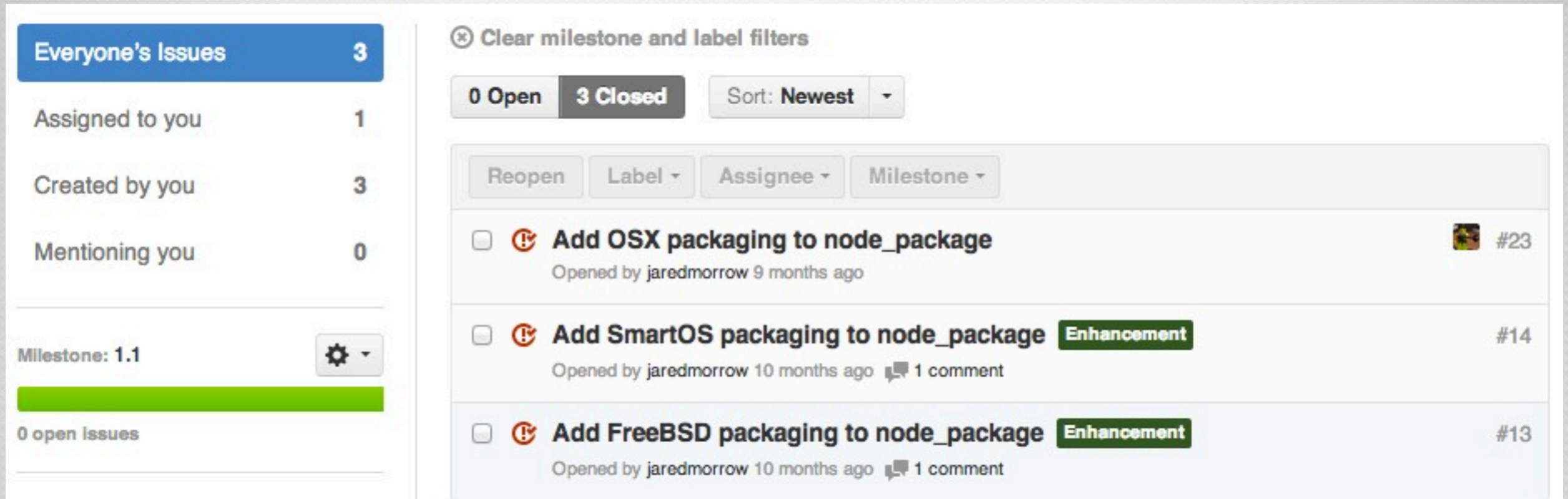
August 2012



**Basho Announces Open Source Riak CS and General  
Availability of Riak CS Enterprise v1.3**

# Riak CS Open Source





Everyone's Issues 3

Assigned to you 1

Created by you 3

Mentioning you 0

Milestone: 1.1

0 open issues

Clear milestone and label filters

0 Open 3 Closed Sort: Newest

Reopen Label Assignee Milestone

- Add OSX packaging to node\_package #23  
Opened by jaredmorrow 9 months ago
- Add SmartOS packaging to node\_package Enhancement #14  
Opened by jaredmorrow 10 months ago 1 comment
- Add FreeBSD packaging to node\_package Enhancement #13  
Opened by jaredmorrow 10 months ago 1 comment

# RiakCS Now On Unix!





Vagabond opened this pull request 4 months ago

Edit

## Switch riak to use node\_package for packaging

No one is assigned

Milestone: 1.4



No description given.

Merged

+ 206 additions

- 4,029 deletions

2 participants



# Finally!





# Repos Using Node Package Today

- Riak (1.4+)
- Riak EE (1.4+)
- RiakCS (all versions)
- Stanchion (a separate app for RiakCS)
- basho\_bench (future, there-is-a-branch-for-that)





# REQUIREMENTS

Nothing is free



# Requirements

- An Erlang app you want to stop and start like a daemon (not a library)
- rebar
- reltool\*\*
- a Makefile\*\*

\*\* This might not be entirely true, just what we use and is tested





# FEATURES

The tool has to actually **do** something for you



# Supported Platforms

- Ubuntu 10.04 -> Current && Debian 6 & 7
- RHEL / Centos 5 & 6 && Fedora 14 -> Current
- Solaris 10u9
- SmartOS datasets 1.6, 1.8, 13.1
- OSX
- FreeBSD 9



# Supported Platforms

- Ubuntu 10.04 -> Current && Debian 6 & 7
- RHEL / Centos 5 & 6 && Fedora 14 -> Current
- Solaris 10u9
- **SmartOS datasets 1.6, 1.8, 13.1**
- OSX
- **FreeBSD 9**



# Package Features

- RPM / Deb based builds have init.d scripts
- RPM / Deb are lint checked
- Complete Package Metadata
- Package dependencies and conflicts handled for most cases
- Sensible default installation paths for all OS's



# Runtime Features

- Familiarity with rebar simplenode template
- Stable runner scripts used in production
- Easy to include env.sh that sets paths and common functions for other scripts to use
- <app>.pid file management for integration w/ other tools
- SMF manifest files for SmartOS (including epmd)







# SHORTCOMINGS

No tool is perfect



# Shortcomings

- FreeBSD / SmartOS should support ports/pkgsrc
- Package Conflicts, Obsoletes are not exposed
- Easy to use `node_package`, hard to develop for it
- No support for `upstart`, `systemd`, etc.
- Basho is the only user (that I know of)





# HOW NODE\_PACKAGE WORKS

I heard you like templates, so I put some templates in your templates



# How node\_package Works

- App rebar brings in node\_package as a dependency
- App Makefile calls into node\_package/Makefile
- node\_package/Makefile determines OS, calls <os>/makefile.bootstrap
- makefile.bootstrap loads <os>.template as overlay
- Package build starts with overlay in place



# How node\_package Works

- App rebar brings in node\_package as a dependency
- App Makefile calls into node\_package/Makefile
- node\_package/Makefile determines OS, calls <os>/makefile.bootstrap
- makefile.bootstrap loads <os>.template as overlay
- Package build starts with overlay in place



```
28 {deps, [  
29   {node_package, "1.2.2", {git, "git://github.com/basho/node_package", {t  
30   {webmachine, ".*", {git, "git://github.com/basho/webmachine", {tag, "1.  
31   {riakc, ".*", {git, "git://github.com/basho/riak-erlang-client", {tag,  
32   {lager, ".*", {git, "git://github.com/basho/lager", {tag, "2.0.0rc2"}}}  
33   {eper, ".*", {git, "git://github.com/basho/eper.git", "3280b736057a6cf5  
34   {sext, ".*", {git, "git://github.com/esl/sext", {tag, "0.4.1"}}},  
35   {druuid, ".*", {git, "git://github.com/kellymclaughlin/druuid.git", {ta
```

your\_app/rebar.config



# How node\_package Works

- App rebar brings in node\_package as a dependency
- **App Makefile calls into node\_package/Makefile**
- node\_package/Makefile determines OS, calls <os>/makefile.bootstrap
- makefile.bootstrap loads <os>.template as overlay
- Package build starts with overlay in place



```
182 dist: package.src
183     cp package/${PKG_ID}.tar.gz .
184
185 package: package.src
186     make -C package -f ${PKG_ID}/deps/node_package/Makefile
187
188 pkgclean: distclean
189     rm -rf package
```

your\_app/Makefile





# How node\_package Works

- App rebar brings in node\_package as a dependency
- App Makefile calls into node\_package/Makefile
- node\_package/Makefile determines OS, calls `<os>/makefile.bootstrap`
- makefile.bootstrap loads `<os>.template` as overlay
- Package build starts with overlay in place



```

37 ifeq ($(OS),FreeBSD)
38 OSNAME      = FreeBSD
39 ARCH        = $(shell uname -m)
40 PKGERDIR    = fbsd
41 BUILDDIR    = fbsdbuild
42 endif
43
44 ifeq ($(OS),SunOS)      # Solaris flavors
45 KERNELVER = $(shell uname -v | grep -c joyent 2> /dev/null)
46 ARCH      = $(shell file `which erlc` | grep -c 64-bit 2> /dev/null | awk \
47             '{if ($$1 == "0") {print "i386"} else {print "x86_64"}}')
48
49 ifneq ($(KERNELVER),0)  # SmartOS
50 OSNAME    = SmartOS
51 PKGERDIR  = smartos
52 BUILDDIR  = smartosbuild
53
70 ## Call platform dependent makefile
71 ostype: varcheck setversion
72     $(if $(PKGERDIR),,$(error "Operating system '$(OS)' not supported by node_package"))
73     $(MAKE) -f $(PKG_ID)/deps/node_package/priv/templates/$(PKGERDIR)/Makefile.bootstrap
74

```

node\_package/Makefile



# How node\_package Works

- App rebar brings in node\_package as a dependency
- App Makefile calls into node\_package/Makefile
- node\_package/Makefile determines OS, calls <os>/makefile.bootstrap
- **makefile.bootstrap loads <os>.template as overlay**
- Package build starts with overlay in place



```
7 bootstrap:
8     mkdir -p $(PKG_ID)/fbsd
9     cd $(PKG_ID)/fbsd && $(REBAR) -v create \
10         template_dir=../deps/node_package/priv/templates \
11         template_vars=../pkg.vars.config template=fbsd
12     $(MAKE) -C $(PKG_ID) -f fbsd/Makefile
```

node\_package/priv/templates/fbsd/Makefile.bootstrap



```

4 {variables, [
5     {package_name, "package_name"},
6     {package_install_name, "package_install_name"},
7     {package_install_user, "package_install_user"},
8     {package_install_user_desc, "package_install_user_desc"},
9     {package_install_group, "package_install_group"},
10    {bin_or_sbin, "bin"},
11    {freebsd_package_category, "db"},
12
13    %% Platform-specific installation paths
14    {platform_bin_dir, "/usr/local/{{bin_or_sbin}}"},
15    {platform_data_dir, "/var/db/{{package_install_name}}"},
16    {platform_etc_dir, "/usr/local/etc/{{package_install_name}}"},
17    {platform_base_dir, "/usr/local/lib/{{package_install_name}}"},
18    {platform_lib_dir, "/usr/local/lib/{{package_install_name}}/lib"},
19    {platform_log_dir, "/var/log/{{package_install_name}}"}
20 ]
21 }.
22 {template, "Makefile", "Makefile"}.
23 {template, "vars.config", "vars.config"}.
24 {template, "+COMMENT", "+COMMENT"}.
25 {template, "+DEFINSTALL", "+DEFINSTALL"}.

```

node\_package/priv/templates/fbsd/fbsd.template



# How node\_package Works

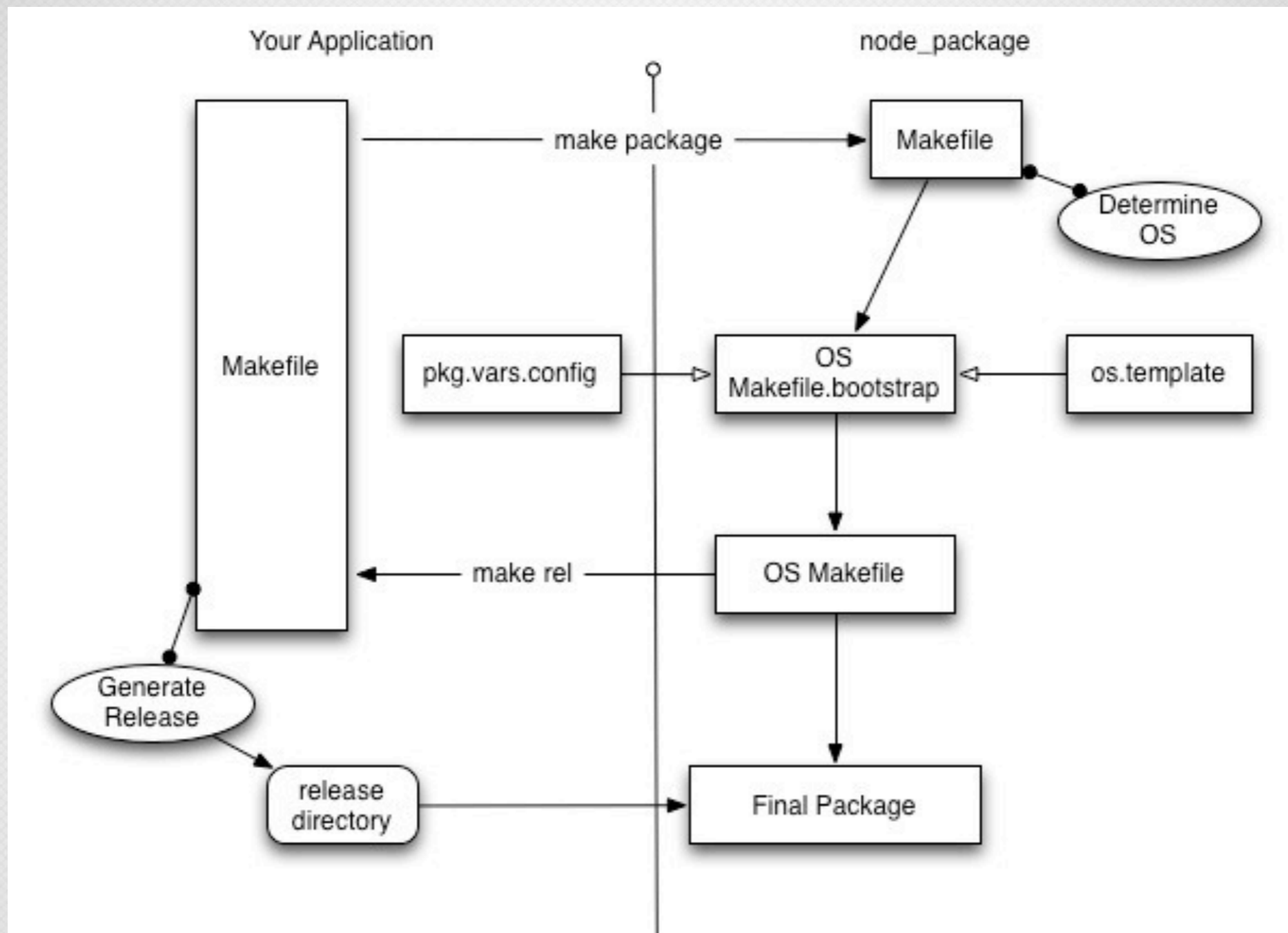
- App rebar brings in node\_package as a dependency
- App Makefile calls into node\_package/Makefile
- node\_package/Makefile determines OS, calls <os>/makefile.bootstrap
- makefile.bootstrap loads <os>.template as overlay
- Package build starts with overlay in place



```
131 $(BUILD_STAGE_DIR): buildrel
132     @echo "Copying rel directory to staging directory"
133     mkdir -p $@
134     mkdir -p $(PBIN_DIR)
135     cp -R rel/{{package_install_name}}/bin/* $(PBIN_DIR)
136     mkdir -p $(PETC_DIR)
137     cp -R rel/{{package_install_name}}/etc/* $(PETC_DIR)
138     mkdir -p $(PLIB_DIR)
139     cp -R rel/{{package_install_name}}/lib $(PLIB_DIR)
140     cp -R rel/{{package_install_name}}/erts-* $(PLIB_DIR)
141     cp -R rel/{{package_install_name}}/releases $(PLIB_DIR)
142     mkdir -p $(PDATA_DIR)
143     cp -R rel/{{package_install_name}}/data/* $(PDATA_DIR)
144     touch $(PDATA_DIR)/.empty
145     mkdir -p $(PLOG_DIR)
146     touch $(PLOG_DIR)/.empty
147
148
149     # Build the release we need to package
150     # * Ensure all binaries are executable
151     # * copy the vars.config over for build config
152     buildrel:
153     OVERLAY_VARS="overlay_vars=../fbsd/vars.config" $(MAKE) deps rel
154     chmod 0755 rel/{{package_install_name}}/bin/* rel/{{package_install_name}}/erts-*/bin/*
```

node\_package/priv/templates/fbsd/Makefile









# WALKTHROUGH WITH STANCHION



# PKG.VARS.CONFIG

Core package settings



```
4 %%  
5 %% Packaging  
6 %%  
7 {package_name, "stanchion"}.  
8 {package_install_name, "stanchion"}.  
9 {package_install_user, "stanchion"}.  
10 {package_install_group, "riak"}.  
11 {package_install_user_desc, "Stanchion user"}.  
12 {package_shortdesc, "Request Serialization Application"}.  
13 {package_desc, "Request Serialization Application"}.  
14 {package_commands, {list, [[{name, "stanchion"}]]}}.  
15 {package_patch_dir, "basho-patches"}.  
16 {bin_or_sbin, "sbin"}.  
17 {license_type, "Proprietary"}.  
18 {copyright, "2013 Basho Technologies, Inc"}.  
19 {vendor_name, "Basho Technologies, Inc"}.  
20 {vendor_url, "http://basho.com"}.  
21 {vendor_contact_name, "Basho Package Maintainer"}.  
22 {vendor_contact_email, "packaging@basho.com"}.  
23 {license_full_text, "This software is provided under license from Basho Technologies."}.  
24 {solaris_pkgname, "BASHOstanchion"}.
```

<app>/pkg.vars.config





# MAKEFILE

The bootstrap



```
105 .PHONY: package
106 export PKG_VERSION PKG_ID PKG_BUILD BASE_DIR ERLANG_BIN REBAR OVERLAY_VARS RELEASE
107
108 package.src: deps
109     mkdir -p package
110     rm -rf package/$(PKG_ID)
111     git archive --format=tar --prefix=$(PKG_ID)/ $(PKG_REVISION) | (cd package && tar -xf -)
112     ${MAKE} -C package/$(PKG_ID) deps
113     for dep in package/$(PKG_ID)/deps/*; do \
114         echo "Processing dep: $$dep"; \
115         mkdir -p $$dep/priv; \
116         git --git-dir=$$dep/.git describe --tags >$$dep/priv/vsn.git; \
117     done
118     find package/$(PKG_ID) -depth -name ".git" -exec rm -rf {} \;
119     tar -C package -czf package/$(PKG_ID).tar.gz $(PKG_ID)
120
121 dist: package.src
122     cp package/$(PKG_ID).tar.gz .
123
124 package: package.src
125     ${MAKE} -C package -f $(PKG_ID)/deps/node_package/Makefile
```

<app>/Makefile





# VAR.S.CONFIG

additions to support notion of local devrels



```
30 %%
31 %% bin/stanchion
32 %%
33 {data_dir,      "{{target_dir}}/data"}.
34 {runner_script_dir, "$(cd ${0%/*} && pwd)"}.
35 {runner_base_dir, "{{runner_script_dir}}/.."}.
36 {runner_etc_dir, "$RUNNER_BASE_DIR/etc"}.
37 {runner_log_dir, "$RUNNER_BASE_DIR/log"}.
38 {runner_lib_dir, "$RUNNER_BASE_DIR/lib"}.
39 {runner_patch_dir, "$RUNNER_BASE_DIR/lib/basho-patches"}.
40 {pipe_dir,      "/tmp/$RUNNER_BASE_DIR/"}.
41 {runner_user,   ""}.
42 {runner_wait_process, "stanchion_server_sup"}.
```

<app>/rel/vars.config \*not required for pkgs\*



# RELTOOL.CONFIG

Don't worry, it'll be fine





```
40 {overlay, [  
41     %% Scan for scripts in included apps  
42     %% Setup basic dirs  
43     {mkdir, "log"},  
44     {mkdir, "data/stanchion"},  
45  
46     %% Copy base files for starting and interacting w/ node  
47     {copy, "../deps/node_package/priv/base/erl",  
48         "{{erts_vsn}}/bin/erl"},  
49     {copy, "../deps/node_package/priv/base/nodetool",  
50         "{{erts_vsn}}/bin/nodetool"},  
51     {template, "../deps/node_package/priv/base/runner",  
52         "bin/stanchion"},  
53     {template, "../deps/node_package/priv/base/env.sh",  
54         "lib/env.sh"},
```

<app>/rel/reltool.config



ONLY FOUR FILES CHANGED



# What Did That Buy Us?

- 'make rel' still works (vars.config, reltool.config, minor changes)
- 'make devrel' still works (5-node riak cluster)
- Packages override all our vars.config settings
- Even if we don't **need** packages for X, we have that ability





# WHAT DOES IT LOOK LIKE

too scared to live demo, sorry



# Package Metadata

```
jared@fresh-debian-6:~$ dpkg-deb --info riak_1.4.0pre1-1_amd64.deb
new debian package, version 2.0.
size 26850684 bytes: control archive= 1367 bytes.
    93 bytes,      5 lines      conffiles
   428 bytes,     11 lines      control
  1028 bytes,     54 lines      * postinst      #!/bin/sh
  1797 bytes,     68 lines      * postrm        #!/bin/sh
Package: riak
Version: 1.4.0pre1-1
Architecture: amd64
Maintainer: Basho Package Maintainer <packaging@basho.com>
Installed-Size: 46332
Depends: libc6 (>= 2.3.4), libgcc1 (>= 1:4.1.1), libncurses5 (>= 5.7+20100313),
libssl0.9.8 (>= 0.9.8m-1), libstdc++6 (>= 4.4.0), adduser, logrotate, sudo
Section: net
Priority: extra
Homepage: http://basho.com
Description: Riak is a distributed data store
 Riak is a distributed data store
```



# Package Install

```
jared@fresh-debian-6:~$ sudo dpkg -i riak_1.4.0pre1-1_amd64.deb
```

```
Selecting previously deselected package riak.
```

```
(Reading database ... 25988 files and directories currently installed.)
```

```
Unpacking riak (from riak_1.4.0pre1-1_amd64.deb) ...
```

```
Setting up riak (1.4.0pre1-1) ...
```

```
Adding group `riak' (GID 106) ...
```

```
Done.
```

```
Adding system user `riak' (UID 104) ...
```

```
Adding new user `riak' (UID 104) with group `riak' ...
```

```
Not creating home directory `/var/lib/riak'.
```

```
Processing triggers for man-db ...
```



# Runner Options

```
jared@fresh-debian-6:~$ riak
Usage: riak {start | stop| restart | reboot | ping | console | attach |
          attach-direct | ertspath | chkconfig | escript | version |
          getpid | top [-interval N] [-sort reductions|memory|msg_q] [-
lines N] }
jared@fresh-debian-6:~$ riak version
1.4.0pre1
jared@fresh-debian-6:~$ sudo riak start
jared@fresh-debian-6:~$ riak ping
pong
jared@fresh-debian-6:~$ riak getpid
1664
jared@fresh-debian-6:~$ cat /var/run/riak/riak.pid
1664
```



# Separation Of Privilege

```
jared@fresh-debian-6:~$ riak ping
pong
jared@fresh-debian-6:~$ riak getpid
1664
jared@fresh-debian-6:~$ /etc/init.d/riak
Usage: /etc/init.d/riak {start|stop|ping|restart|force-reload|status}
jared@fresh-debian-6:~$ /etc/init.d/riak status
pong
riak is running
jared@fresh-debian-6:~$ riak attach
Unable to access /var/run/riak, permission denied, run script as root
jared@fresh-debian-6:~$ riak stop
Unable to access /var/run/riak, permission denied, run script as root
jared@fresh-debian-6:~$ sudo riak stop
ok
```





# Common Shell Library

```
riak@fresh-debian-6:~$ cat /usr/lib/riak/lib/env.sh
#!/bin/sh
# -*- tab-width:4;indent-tabs-mode:nil -*-
# ex: ts=4 sw=4 et

# installed by node_package (github.com/basho/node_package)

RUNNER_SCRIPT_DIR=/usr/bin
RUNNER_SCRIPT=${0##*/}

RUNNER_BASE_DIR=/usr/lib/riak
RUNNER_ETC_DIR=/etc/riak
RUNNER_LOG_DIR=/var/log/riak
RUNNER_LIB_DIR=/usr/lib/riak/lib
RUNNER_PATCH_DIR=/usr/lib/riak/lib/basho-patches
PIPE_DIR=/tmp/riak/
RUNNER_USER=riak
APP_VERSION=1.4.0pre1

<snip>
...
```



# Easy To Add App Specific Scripts

```
riak@fresh-debian-6:~$ head -6 `which riak-admin`
#!/bin/sh
# -*- tab-width:4;indent-tabs-mode:nil -*-
# ex: ts=4 sw=4 et

# Pull environment for this install
. "/usr/lib/riak/lib/env.sh"

<snip>
...

riak@fresh-debian-6:~$ riak-admin
Usage: riak-admin { cluster | join | leave | backup | restore | test |
                  reip | js-reload | erl-reload | wait-for-service |
                  ringready | transfers | force-remove | down |
                  cluster-info | member-status | ring-status | vnode-status |
                  aae-status | diag | status | transfer-limit | reformat-indexes |
```



# Sensible Uninstall

```
jared@fresh-debian-6:~$ sudo dpkg -r riak
(Reading database ... 28211 files and directories currently installed.)
Removing riak ...
Processing triggers for man-db ...
jared@fresh-debian-6:~$ ls /var/lib/riak/
anti_entropy  bitcask  kv_vnode  ring
jared@fresh-debian-6:~$ ls /var/log/riak/
console.log  crash.log  erlang.log.1  error.log  run_erl.log
jared@fresh-debian-6:~$ ls /etc/riak/
app.config  cert.pem  key.pem  vm.args

jared@fresh-debian-6:~$ sudo dpkg --purge riak
(Reading database ... 25997 files and directories currently installed.)
Removing riak ...
Purging configuration files for riak ...
Removing user `riak' ...

jared@fresh-debian-6:~$ ls /etc/riak
ls: cannot access /etc/riak: No such file or directory
jared@fresh-debian-6:~$ ls /var/lib/riak
ls: cannot access /var/lib/riak: No such file or directory
```



# QUESTIONS?

Be gentle please.

[stanchion-build-debian-6-64](#)

#201  
build successful

[stanchion-build-debian-7-64](#)

#21  
build successful

[stanchion-build-fedora-17-64](#)

#201  
build successful

[stanchion-build-freebsd-9-64](#)

#127  
build successful

[stanchion-build-osx-108-64](#)

#23  
build successful

[stanchion-build-smartos-16-64](#)

#21  
build successful

[stanchion-build-smartos-18-64](#)

#107  
build successful

[stanchion-build-smartos-131-64](#)

#21  
build successful

[stanchion-build-solaris-10u9-64](#)

#109  
build successful

[stanchion-build-ubuntu-1004-64](#)

#200  
build successful

[stanchion-build-ubuntu-1104-64](#)

#200

