

PyBOMBS & CGRAN — An Introduction

Martin Braun

GNU Radio Conference 2016

Outline

- 1 How do I even get started with GNU Radio?
- 2 PyBOMBS concepts
- 3 Out-of-tree Modules
- 4 Conclusion

Let's assume for just one minute...

- ...that I have no idea how to get started with GNU Radio.

Let's assume for just one minute...

- ... that I have no idea how to get started with GNU Radio.
- I mean, I literally don't even know how to run the examples and tutorials.

Let's assume for just one minute...

- ... that I have no idea how to get started with GNU Radio.
- I mean, I literally don't even know how to run the examples and tutorials.
- No worries. We've got you covered.

Top 4 easiest ways to install GNU Radio

1 The GNU Radio Live DVD



Top 4 easiest ways to install GNU Radio

- 1 The GNU Radio Live DVD
- 2 `apt-get install gnuradio` — use your package manager, whatever that is



Top 4 easiest ways to install GNU Radio

- 1 The GNU Radio Live DVD
- 2 `apt-get install gnuradio` — use your package manager, whatever that is
- 3 PyBOMBS



Top 4 easiest ways to install GNU Radio

- 1 The GNU Radio Live DVD
- 2 `apt-get install gnuradio` — use your package manager, whatever that is
- 3 PyBOMBS
- 4 Source Builds



What you will need

- A Linux or Mac box (one day we'll have Windows covered, but not today)
- Fedora, Ubuntu, CentOS, Debian, and probably many other distros fine
- Python 2.7

What you will need

- A Linux or Mac box (one day we'll have Windows covered, but not today)
- Fedora, Ubuntu, CentOS, Debian, and probably many other distros fine
- Python 2.7
- And what else...

What you will need

- A Linux or Mac box (one day we'll have Windows covered, but not today)
- Fedora, Ubuntu, CentOS, Debian, and probably many other distros fine
- Python 2.7
- And what else...
- Nope, that's it!

Getting started with PyBOMBS

- Install PyBOMBS: `pip install pybombs`
- Optionally install PyBOMBS GUI: `pip install pybombs-qtgui`
- Next, run some magic commands that I'll skip for now.
- Let's just install GNU Radio and all major dependencies:
- `pybombs prefix init -a default -R gnuradio-default /home/mbr0wn/prefix/gnuradio`

Getting started with PyBOMBS

- Install PyBOMBS: `pip install pybombs`
- Optionally install PyBOMBS GUI: `pip install pybombs-gtgui`
- Next, run some magic commands that I'll skip for now.
- Let's just install GNU Radio and all major dependencies:
- `pybombs prefix init -a default -R gnuradio-default /home/mbr0wn/prefix/gnuradio`
- Whoa, slow down hot shot. . .

Outline

- 1 How do I even get started with GNU Radio?
- 2 PyBOMBS concepts**
- 3 Out-of-tree Modules
- 4 Conclusion

Why do I need a *prefix*?

- Standard Unix way (since probably even Marcus Leech started) is to install software either into `/usr` or `/usr/local`.
- Self-compiled software would default into the latter
- This is simple and understood, but has some problems:
 - No standard way to uninstall hand-installed software
 - Only one installation at a time possible
 - Requires admin (sudo) privileges
- So how about we install it into some random directory in our home?
- `/home/mbr0wn/prefix/gnuradio` — The prefix!
- Solves all the problems above, except we now need to tell the system where to look for stuff (*environment variables* need setting)

pybombs prefix init

- Show me that command again:

```
pybombs  
prefix init  
--alias default  
--recipe gnuradio-default  
/home/mbr0wn/prefix/gnuradio
```

- Aah, I get it now. We install into `prefix/gnuradio`, and we use the rules defined in `gnuradio-default`.
- We call such definitions a *recipe*.

So... recipes?

- Remember those magic commands I skipped? Here they are:

```
$ pybombs recipes add gr-recipes  
git+https://github.com/gnuradio/gr-recipes.git  
$ pybombs recipes add gr-etccetera  
git+https://github.com/gnuradio/gr-etccetera.git
```

- These are part of most people's installation. (See PyBOMBS manual).
- Will install all default recipes.
- Let's take a quick look at one of those.

How do I install things then?

- `pybombs install [-p PREFIX] PACKAGE`
- Let me show you.
- `PREFIX` is either an alias, or a full path.

And how do I run GNU Radio now?

- First, initialize your prefix of choice:
- `source $PREFIX/setup_env.sh`
- You're good. Run: `gnuradio-companion` if you don't believe me.
- You can initialize any number of prefixes at once (only one per shell, though).
- Clicky launchers also work fine.

Why is PyBOMBS accessing my system?

- PyBOMBS is not a container replacement.
- It will use any means to get your system ready to build software (e.g., GNU Radio).
- Your system's package manager is one of those means.
- It might need elevated privileges. This is normal.
- Here's a nice trick: `pybombs install --deps-only gnuradio`

Outline

- 1 How do I even get started with GNU Radio?
- 2 PyBOMBS concepts
- 3 Out-of-tree Modules**
- 4 Conclusion

But what is there to install?

- You could browse the list of recipes.
- But let's have something nice instead!
- `www.cgran.org` FTW!
- Most the things listed on CGRAN are *out-of-tree modules*.

Out-of-tree Modules

- GNU Radio comes with a lot of blocks, but a lot of the value comes from the ability to extend it.
- Other people like to share their extensions to GNU Radio!
- Side note: Making OOTs is really easy. Just use `gr_modtool`!
- See:
gnuradio.org/redmine/projects/gnuradio/wiki/Guid

PSA for content producers

- If you're a producer of GNU Radio-related products or projects, listen up!
- Please consider using PyBOMBS.
- We're trying to consolidate installation methods.
- Create your own recipe repository if you have anything else than a single standard recipe.
- Include manifests in your OOTs so we can source them for CGRAN.

Outline

- 1 How do I even get started with GNU Radio?
- 2 PyBOMBS concepts
- 3 Out-of-tree Modules
- 4 Conclusion**

FAQ

- Q: Why not just use the package manager?
A: You can use both. PyBOMBS will give you the most up-to-date code, and works for most OOTs.
- Q: Can't I just do this all by hand?
A: Knock yourself out!
- Q: Will PyBOMBS help me cross-compile? A: It'll basically do all the work for you—if you have proper recipes.
- Q: Will you make PyBOMBS work on my system/OS? A: Yes, by gladly accepting your patches!
- Q: Bugfixes/contributions? A: github!
- Q: Why didn't you use an existing tool instead of writing PyBOMBS? A: Many good reasons.

Conclusion

- PyBOMBS is awesome and can save you time.
- Sysadmin-friendly (I believe).
- All our cool stuff is on CGRAN. You want to be on CGRAN, too.
- Thanks to all contributors, users and bugfixers!