

Dependency Management with Carton

Jason Crome

What is Carton?

- Top-notch dependency management
- Can specify a specific range of allowed versions, or a specific one. **This is huge!**
- Runs your app with only the modules you specify
- Super-easy to deploy and distribute your app with no module/dependency hell

Why's this useful?

- Consider Tau Station (a game I worked on)
- Developer stymied by test failures
- Only one on team using `Test::Class::Moose 0.77`
- Spend inordinate amount of time tracking down to find it was a version issue
- Better dependency/module management would have made this a non-issue

How Carton works

- Uses `cpanfile` to specify versions
- Can create a `cpanfile.snapshot` to pin specific versions that are currently installed
- Distribute these to other machines with your application to deploy/run your app

What's in cpanfile?

```
# This version and any after
requires "Dancer2", ">= 0.206000";

# No specific version needed
requires "Dancer2::Plugin::Database" => 0;

on "test" => sub {
    requires "Test::More" => "0";
    requires "Test::Class::Moose", ">= 0.73, < 0.77";
};
```

Installing Carton

```
# From CPAN  
cpanm Carton
```

Using Carton

```
# Install module version listed in cpanfile from CPAN  
carton install
```

```
# Modules are installed to local/  
# Version info captured to cpanfile.snapshot
```

```
# Commit to github  
git add cpanfile.snapshot  
git commit -m"Capture module version snapshot."  
git push
```

```
# Clone to other machine, then  
carton install --deployment
```

```
# Now, run your app  
carton exec plackup sbin/app.psgi
```

Bundling with Carton

- `carton bundle` bundles the tarballs for your modules and their dependencies into a directory
- Easy to distribute across a local network
- Deploy via `carton install --cached`
- Can install without CPAN or an internet connection

A final note

- Carton creates two things in your project directory:
`local/` and `.carton`
- These should **not** be distributed with your app
- Make it easy: add to `.gitignore`

Questions?

Thank you!



Copyright 2019, Jason A. Crome