LEARNING FUCKING HASKELL WITH NETWORK SHIT

Thomas Gebert

WILEY-INTERSCIENCE

CONTENTS

CHAPTER 1

GETTING SHIT INSTALLED

1.1 Introduction

The shit that's always the hardest fucking part of getting into a langauge is getting the compiler and tooling set up. Well, no, that's a fucking lie, there are lots of parts that are probably harder, but by the time you get to them you're probably already committed to the language so there's no use in complaining.

1.1.1 GHC

The most important thing you'll need to do Haskell development is The Glorious Haskell Compiler, or GHC for short. There are other Haskell compilers out there, like YHC or UHC, but I don't know anyone that uses them, and I can't be bothered to try them out myself. When people say "Haskell", they usually seem to mean "GHC"

1.1.2 Bullshit about GHC

While GHC is what you need to to compile your Haskell program, it can be pretty frustrating between different point versions, and especially irritating between different platforms. For the love of God, do *not* just install GHC from the command line in your favorite operating system's package manager, since this will almost certainly be wrong, and confuse you.

There's also the fact that GHC is *only* the compiler. Handling dependencies, managing the project files, and handling information about the project itself.

To make sure it's done correctly, you'll need to install either the Haskell Platform (the wrong way), or Stack (the much better way).

1.2   The Shitty Old Way

Back in the old days, back in my youth, the main way to get Haskell installed on your machine was to download and install the Haskell Platform.

The Haskell Platform includes GHC, Cabal, and a bunch of the most commonly used packages.

1.2.1   What the fuck is Cabal?

As mentioned before, GHC is *just* a compiler. I'm not saying that it's a bad thing, it's just limited in scope. Cabal was the first real attempt to help manage projects.

But there's not much point to doing that shit anymore since they finally solved all these problems and invented the Stack application. At this point you really can just say "Fuck it, just install Stack".

1.3   What the hell is Stack?

Stack is a Haskell program by FPComplete to make it easier to make other Haskell programs.

You'll actually find that to be a bit of a pattern in the Haskell universe. It feels like half of all the programs for it are just there to make you feel cool because you're using a Haskell program, or to try to talk you into making more Haskell programs

Anyway, what exactly does Stack do? In a nutshell: it manages your Haskell your entire Haskell project.

Using Stack, you can have your dependencies, structure, compilation, and even your compiler. In the process, you get something that man had previously only dreamed of: reproducible builds.

This is new shit, since it used to be really fucking difficult to get different versions of GHC installed on the same machine, especially if you decided it would be a good idea to use a Linux package manager.

1.4 Installing Stack

Ok, enough rambling about this Stack shit, I'm getting tired of talking about it anyway.

To go about installing it on a sane Mac or Linux platform, you should be able to do something to the effect of

```
curl -sSL https://get.haskellstack.org/ | sh
```

If you're on a shitty operating system like Windows, you don't have access to tools that were invented in the last forty years like "pipe". I guess it's just too advanced for Seattle.

What you'll need to do is just go to the haskellstack.org website, and download the Windows installer.

By the time you read it, this has probably changed, so you'll have to look up the updated URL anyway. I don't really know why I decided to use a printed medium to write a book about a fucking functional language that's hellbent on constantly updating. It's just going to get outdated.

1.5 Using Stack

So now you should have stack installed, and all its glory is available to you. Unless you installed it wrong. Don't exclude that possibility, it will be your downfall.