Hmm...
Haskell Modular Mindset

• Haskell truly is modular – that’s great!

• Being precise about modularity helps inter-op.

• Modularization is cheap!

• Haskell 2010 is NOT monolithic – we just haven’t explored the modules yet.
module Foo where

import C

foo :: (C a b) => a -> b
foo = ...
MPTCs and GHC

• With MultiParamTypeClasses on (and only then), GHC allows declaration of
  – Classes with multiple parameters
  – Instances of classes with multiple parameters

• Without MultiParamTypeClasses, GHC still allows constraints mentioning classes with multiple parameters
  – No flags needed, but not Haskell 2010!
My solution

1. Define: MultiParamTypeClasses, as GHC.
2. Define: MultiParamConstraints, enables mentioning MPTCs in class constraints only.
3. Let MultiParamTypeClasses imply/subsume MultiParamConstraints.
4. Have GHC enable MultiParamConstraints by default.

This makes the discrepancy clear, and gives other tools (e.g. haskell-src-exts) the ability to simulate GHC’s behavior, without ad-hoc ugliness.
I LOVE GHC!