Traits: Evaluating a Language Feature
Object-Oriented Candyland or Inheritance Quagmire?

Emerson Murphy-Hill, Student Researcher, The Evergreen State College
Andrew P. Black, Research Collaborator, OGI School of Science and Engineering
Judy Cushing, Faculty Sponsor, The Evergreen State College

Traits are designed to be the building blocks of classes.
Traits have some of the benefits of multiple inheritance.
Traits allow us to reuse code on a fine-grained level.
Traits are flexible and allow for static or dynamic selection.

Implementation

Abstract Classes

Traits are designed to be the building blocks of classes.

Single Inheritance

String - Slow and Inefficient

Ropes - Faster and More Efficient

Conclusions

Traits Demand Reliable Methods
Traits have Unanticipated Side Effects

Results

Traits allowed 46% of the methods to be reused!

The End

Start!

Single Inheritance With traits

A solution: methods which came from ropes - faster and more efficient

Implementation

TRectangle + TStringImpl + TString

Concat

W

Abstract Classes

Traits are designed to be the building blocks of classes.

Traits have some of the benefits of multiple inheritance.
Traits allow us to reuse code on a fine-grained level.
Traits are flexible and allow for static or dynamic selection.

Single Inheritance

String - Slow and Inefficient

Ropes - Faster and More Efficient

Conclusions

Traits Demand Reliable Methods
Traits have Unanticipated Side Effects

Results

Traits allowed 46% of the methods to be reused!

The End

Start!