decltrait

Tian Liao

December 29, 2023

Document number:
Date: December 29, 2023
Audience:
Authors: Tian Liao, Mingxin Wang
Reply-to: Tian Liao <tilia@microsoft.com>

1 Introduction

2 Proposal

2.1 Language feature

```
struct Drawable { void print(); };
struct Rectangle { void print() { std::println("Rectangle."); } };
struct Circle { void print() { std::println("Circle."); } };

void foo() {
  auto print_fn = decltrait(Drawable, Rectangle)->print;
  Rectangle rectangle;
  print_fn(&rectangle); // prints "Rectangle.".

  print_fn = decltrait(Drawable, Circle)->print;
  Circle circle;
  print_fn(&circle); // prints "Circle.".

  print_fn = decltrait(Drawable, &rectangle)->print;
  print_fn(); // prints "Rectangle.".

  static_assert(std::is_pointer<
    decltrait(Drawable, Rectangle)>::value); // true.
  static_assert(!std::is_same_v<
    Drawable*,
    decltype(decltrait(Drawable, Circle))>); // true (not the same).
  static_assert(std::is_same_v<
    decltype(decltrait(Drawable, Rectangle)),
    decltype(decltrait(Drawable, Circle))>); // true.

  static_assert(std::is_pointer<
    decltrait(Drawable, &rectangle)>::value); // true.
  static_assert(!std::is_same_v<
    Drawable*,
    decltype(decltrait(Drawable, &rectangle))>); // true (not the same).
  static_assert(!std::is_same_v<
    decltype(decltrait(Drawable, Rectangle)),
    decltype(decltrait(Drawable, &rectangle))>); // true (not the same).
  static_assert(std::is_same_v<
    decltype(decltrait(Drawable, &rectangle)),
    decltype(decltrait(Drawable, &circle))>); // true.
}
```

3 Motivation