

fn_traits

The tracking issue for this feature is #29625

See Also: `unboxed_closures`

The `fn_traits` feature allows for implementation of the `Fn*` traits for creating custom closure-like types.

```
#![feature(unboxed_closures)]
#![feature(fn_traits)]

struct Adder {
    a: u32
}

impl FnOnce<(u32, )> for Adder {
    type Output = u32;
    extern "rust-call" fn call_once(self, b: (u32, )) -> Self::Output {
        self.a + b.0
    }
}

fn main() {
    let adder = Adder { a: 3 };
    assert_eq!(adder(2), 5);
}
```