

# Move Semantics in Rust

Programming Research Group, Winter 2015

Michael McGirr

Oregon State University

Graduate School of Electrical Engineering and Computer Science

1. There is only ever *exactly one* binding to any given resource.
2. A variable owns what it is bound to.
3. When a variable's lifetime goes out of scope the bound resources are freed.
4. We may transfer ownership to something else. The moved values are then owned by something else.
5. A binding may be able to borrow a resource from another binding and use it before it returns it to the original scope.
6. An owner may be able to mutate a resource as long as it has not been lent out.
7. An borrower may be able to mutate a resource that it borrows as long as it has not been lent out to any other variables. Only one mutable borrow of a resource at a time.
8. More than one binding may borrow a resource but they may not mutate it.