**Method sets**

A type may have a *method set* associated with it (§[Interface types](http://192.168.206.128:6060/ref/spec#Interface_types), §[Method declarations](http://192.168.206.128:6060/ref/spec#Method_declarations)). The method set of an [interface type](http://192.168.206.128:6060/ref/spec#Interface_types) is its interface. The method set of any other type T consists of all methods with receiver type T. The method set of the corresponding pointer type \*T is the set of all methods with receiver \*T or T (that is, it also contains the method set of T). Further rules apply to structs containing anonymous fields, as described in the section on [struct types](http://192.168.206.128:6060/ref/spec#Struct_types). Any other type has an empty method set. In a method set, each method must have a [unique](http://192.168.206.128:6060/ref/spec#Uniqueness_of_identifiers)[method name](http://192.168.206.128:6060/ref/spec#MethodName).

译：一个type类型有一个方法集合与之相关。interface 类型的方法集就是接口本身。

type T 的method set 是所有receiver type为 T的mothod集合；

type \*T的method set 是所有receiver type为 T或者\*T的mothod集合；

对于sturct中嵌入匿名Field字段的更深层次规则参考 struct type 章节。任何其他type 拥有个空的method set, 在一个method set中，任何method的名称必须唯一。

The method set of a type determines the interfaces that the type [implements](http://192.168.206.128:6060/ref/spec#Interface_types) and the methods that can be [called](http://192.168.206.128:6060/ref/spec#Calls) using a receiver of that type.

译：一个type 类型的 method set  决定该type所实现的接口（换句话该type的 method set是接口 method set 的超集，则该type implemnet 改interface），以及该类型作为receiver可以调用的方法集（type 只能调用其method set上的方法：这里的调用模式是Method Expression调用模式，不包括用type value 或 pointer变量调⽤用所有绑定的方法，因为编译器针对type value 或 pointer变量自动进⾏行类型转换。）。

**struct 匿名字段⽅方法集规则:**

• 如果 S 嵌⼊入匿名类型 T，则 S ⽅方法集包含 T ⽅方法集。

• 如果 S 嵌⼊入匿名类型 \*T，则 S ⽅方法集包含 \*T 的⽅方法集 (T + \*T)。

• 如果 S 嵌⼊入匿名类型 T 或 \*T，则 \*S ⽅方法集包含 \*T 的⽅方法集 (T + \*T)。