

静态语言：

对于在静态语言中定义的类型，变量或类型属性等标识符，通过其IDE提供的Find References功能可以完全找到其所有的被引用点。静态语言完全可以通过静态分析来看懂代码并查找出问题。

动态语言：

对于动态语言，由于其变量类型和类型成员的不确定性，因此静态分析通常会比较困难，这也导致大家通常都认为动态语言写的程序比较难以维护。

动态语言由于其灵活性，如果语言不提供模块化的机制来对系统进行分解，并且没有对全局变量的使用加以约束的话，确实会导致其写成的代码难以分析和维护。

通过为动态语言提供模块化机制不仅可以减少全局变量的使用，而且通过模块化机制也可以隔离系统的复杂性，使得单个模块的功能点相对集中代码量小，使得对于动态语言代码的分析可以基于模块来进行，大大提高了动态语言的静态可分析性。